

Universität Stuttgart



UNIVERSITÄT STUTTGART  
Fakultät 8: Mathematik und Physik  
Institut für Angewandte Analysis und Numerische Simulation

# Efficient Numerical Methods for Total Variation Minimization

Habilitationsschrift

ANDREAS LANGER

Stuttgart 2020



# Contents

<b>Deutsche Zusammenfassung</b>	<b>ix</b>
<b>Outline</b>	<b>xxi</b>
<b>I. Subspace Correction Methods for Total Variation Minimization</b>	<b>1</b>
<b>1. Introduction and Overview</b>	<b>3</b>
<b>2. Subspace Correction Methods for a Class of Nonsmooth and Nonadditive Convex Variational Problems</b>	<b>15</b>
2.1. Introduction . . . . .	15
2.2. Image Restoration with Mixed $L^1/L^2$ Data-Fidelity . . . . .	18
2.2.1. Qualitative Behavior of the $L^1-L^2$ -TV Model . . . . .	18
2.2.2. Practical Behavior . . . . .	20
2.3. Subspace Correction Approach in $L^2(\Omega)$ . . . . .	24
2.4. Subspace Correction Approach for the Discretized Problem . . . . .	26
2.4.1. Notation and Basic Definitions . . . . .	27
2.4.2. Properties of Subspace Correction Methods . . . . .	28
2.5. Application: Domain Decomposition . . . . .	39
2.5.1. Overlapping Domain Decomposition . . . . .	39
2.5.2. Nonoverlapping Domain Decomposition . . . . .	39
2.5.3. Numerical Implementation . . . . .	40
2.5.4. Numerical Experiments . . . . .	44
2.6. Conclusion . . . . .	52
2.7. References . . . . .	52
<b>3. Surrogate Functional Based Subspace Correction Methods for Image Processing</b>	<b>57</b>
3.1. Introduction . . . . .	57
3.2. Notations . . . . .	58
3.3. Subspace Correction Approaches . . . . .	58
3.3.1. Convergence Properties . . . . .	60
3.4. Numerical Experiments . . . . .	62
3.5. References . . . . .	64
<b>4. Nonoverlapping Domain Decomposition Methods for Dual Total Variation Based Image Denoising</b>	<b>65</b>
4.1. Introduction . . . . .	65
4.2. Notations . . . . .	68

## Contents

4.3.	Nonoverlapping Domain Decomposition of the (Pre-)Dual Problem . . . . .	69
4.3.1.	A Parallel Version and Its Convergence . . . . .	74
4.4.	Implementation . . . . .	78
4.5.	Numerical Examples . . . . .	81
4.5.1.	Denoising of 2D Image Data . . . . .	81
4.5.2.	Comparison of the Dual and Primal Domain Decomposition Approach	86
4.5.3.	Denoising 3D Computerized Tomography (CT) Images . . . . .	87
4.6.	References . . . . .	90
<b>5.</b>	<b>Overlapping Domain Decomposition Methods for Total Variation Denoising</b>	<b>93</b>
5.1.	Introduction . . . . .	93
5.2.	Overlapping Domain Decomposition Algorithms . . . . .	96
5.2.1.	Basic Terminology . . . . .	96
5.2.2.	Preliminaries . . . . .	97
5.2.3.	Alternating Algorithm . . . . .	101
5.2.4.	Parallel Algorithm . . . . .	105
5.2.5.	Multi-Subdomain . . . . .	108
5.3.	Subspace Minimization . . . . .	110
5.3.1.	Discretize Before Restriction . . . . .	110
5.3.2.	Restrict Before Discretization . . . . .	113
5.3.3.	Comparison of the Two Restriction Approaches . . . . .	113
5.3.4.	Implementation Issues . . . . .	114
5.4.	Experiments . . . . .	117
5.4.1.	Domain Decomposition . . . . .	117
5.4.2.	Sequential Algorithm . . . . .	118
5.4.3.	Parallel Algorithm . . . . .	119
5.5.	Conclusion . . . . .	123
5.6.	References . . . . .	124
<b>II.</b>	<b>Parameter Selection Methods for Total Variation Models</b>	<b>127</b>
<b>6.</b>	<b>Introduction and Overview</b>	<b>129</b>
<b>7.</b>	<b>Automated Parameter Selection for Total Variation Minimization in Image Restoration</b>	<b>139</b>
7.1.	Introduction . . . . .	139
7.1.1.	Choice of the Fidelity Term . . . . .	140
7.1.2.	Choice of the Scalar Regularization Parameter . . . . .	141
7.1.3.	Spatially Adaptive Parameter . . . . .	142
7.1.4.	Contribution . . . . .	143
7.1.5.	Outline of the Paper . . . . .	144
7.2.	Constrained versus Unconstrained Minimization Problem . . . . .	144
7.3.	Automated Scalar Parameter Selection . . . . .	147
7.3.1.	Statistical Characterization of the Noise . . . . .	147
7.3.2.	Automated Parameter Selection Strategy . . . . .	148

7.4.	Locally Constrained TV Problem . . . . .	152
7.4.1.	Local Filtering . . . . .	153
7.4.2.	Locally Adaptive Total Variation Algorithm . . . . .	155
7.5.	Total Variation Minimization . . . . .	160
7.5.1.	$L^2$ -TV Minimization . . . . .	160
7.5.2.	An Algorithm for $L^1$ -TV Minimization . . . . .	162
7.5.3.	A Primal-Dual Method for $L^1$ -TV Minimization . . . . .	163
7.6.	Numerical Experiments . . . . .	165
7.6.1.	Automatic Scalar Parameter Selection . . . . .	166
7.6.2.	Locally Adaptive Total Variation Minimization . . . . .	170
7.6.3.	Gaussian Noise Removal . . . . .	170
7.6.4.	Deblurring and Gaussian noise removal . . . . .	176
7.6.5.	Impulse Noise Removal . . . . .	177
7.7.	Conclusion and Extensions . . . . .	177
7.8.	References . . . . .	180
<b>8.</b>	<b>Adaptive Regularization for Parseval Frames in Image Processing</b>	<b>185</b>
8.1.	Introduction . . . . .	185
8.2.	Adaptive Regularization Approach . . . . .	186
8.2.1.	The SA-TV Approach . . . . .	187
8.2.2.	Our Approach . . . . .	188
8.3.	Numerical Experiment . . . . .	192
8.3.1.	Reconstruction of Partial Fourier-Data . . . . .	192
8.3.2.	Wavelet Inpainting . . . . .	197
8.4.	Conclusion . . . . .	201
8.5.	References . . . . .	204
<b>9.</b>	<b>Influence of the Regularization Parameter on Box-Constrained Total Variation</b>	<b>207</b>
9.1.	Introduction . . . . .	207
9.2.	Basic Terminology . . . . .	210
9.3.	Limitation of Box-Constrained Total Variation Minimization . . . . .	211
9.4.	A Semismooth Newton Method . . . . .	213
9.4.1.	The Model Problem . . . . .	213
9.4.2.	Dualization . . . . .	214
9.4.3.	Adaptation to Nonscalar $\alpha$ . . . . .	217
9.5.	Numerical Implementation . . . . .	217
9.5.1.	Scalar $\alpha$ . . . . .	218
9.5.2.	Nonscalar $\alpha$ . . . . .	219
9.6.	Numerical Experiments . . . . .	219
9.6.1.	Dependency on the Parameter $\eta$ . . . . .	221
9.6.2.	Box-Constrained versus Non-Box-Constrained . . . . .	221
9.6.3.	Comparison with Optimal Regularization Parameters . . . . .	222
9.7.	Automated Parameter Selection . . . . .	237
9.7.1.	Numerical Examples . . . . .	239
9.8.	Conclusion . . . . .	240
9.9.	References . . . . .	240

## Contents

<b>10. Automated Parameter Selection in the <math>L^1-L^2</math>-TV Model for Removing Gaussian plus Impulse Noise</b>	<b>245</b>
10.1. Introduction . . . . .	245
10.2. Statistical Characterization of the Noise . . . . .	248
10.3. Constrained versus Unconstrained Problem . . . . .	251
10.3.1. Existence of Minimizers . . . . .	251
10.3.2. Stability of the $L^1-L^2$ -TV Model with Respect to Its Parameters . . . . .	254
10.3.3. Further Properties of the $L^1-L^2$ -TV Model . . . . .	256
10.4. Automated Parameter Selection . . . . .	260
10.4.1. Uzawa's Method . . . . .	261
10.4.2. The pAPS-Algorithm . . . . .	262
10.4.3. Solving the Constrained Problem . . . . .	266
10.5. An Algorithm for Solving the $L^1-L^2$ -TV Model . . . . .	267
10.6. Numerical Experiments . . . . .	271
10.6.1. Initial Value $\alpha_i^{(0)}$ . . . . .	272
10.6.2. Gaussian plus Impulse Noise . . . . .	273
10.6.3. Reconstruction of Blurred and Noisy Images . . . . .	276
10.7. Conclusion . . . . .	278
10.8. References . . . . .	279
<b>11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise</b>	<b>287</b>
11.1. Introduction . . . . .	287
11.1.1. Locally Constrained Minimization Problem . . . . .	289
11.2. Locally Adaptive Algorithm . . . . .	290
11.2.1. The Locally Constrained Problem . . . . .	291
11.2.2. On the Choice of the Local Bounds . . . . .	292
11.2.3. Local Parameter Selection . . . . .	293
11.2.4. Primal-Dual Algorithm for Locally Adaptive Total Variation . . . . .	295
11.3. Numerical Results . . . . .	295
11.3.1. Dependency on the Window-Size . . . . .	296
11.3.2. Comparison with Scalar Parameter . . . . .	297
11.3.3. Separated Noise . . . . .	301
11.3.4. Nonhomogeneous Noise . . . . .	302
11.4. References . . . . .	306
<b>III. Finite Element Discretization of the Total Variation</b>	<b>309</b>
<b>12. Introduction and Overview</b>	<b>311</b>
<b>13. Using DUNE-ACFem for Nonsmooth Minimization of Bounded Variation Functions</b>	<b>315</b>
13.1. Introduction . . . . .	315
13.2. Problem Formulation . . . . .	316
13.2.1. Discretization . . . . .	316
13.2.2. Primal-Dual Algorithm . . . . .	318

13.3.	Implementation Details	320
13.3.1.	On DUNE-ACFEM	320
13.3.2.	Image Read-In and Noise Simulation	322
13.3.3.	Implementation of the Primal-Dual Algorithm	323
13.3.4.	Adaptive Refinement	324
13.4.	Numerical Experiments	325
13.4.1.	Comparison of Discrete Spaces	325
13.4.2.	Image Denoising	326
13.4.3.	Strong Scaling	327
13.5.	Summary and Outlook	328
13.6.	Appendix A - Installation Instructions	329
13.7.	Appendix B - Parameters	330
13.8.	References	331



# Deutsche Zusammenfassung

In dieser Arbeit werden neue effiziente numerische Methoden für die Minimierung der totalen Variation und deren theoretische und numerische Analyse präsentiert. Bei der Entwicklung solcher Methoden konzentrieren wir uns auf drei Klassen von Verfahren, nämlich Unterraum-Korrekturverfahren, Parameterwahlmethoden und Finite Elemente Methoden. In diesem Sinn besteht diese Arbeit aus den folgenden drei Teilen:

- I. Unterraum-Korrekturverfahren für die Minimierung der totalen Variation
- II. Parameterwahlmethoden für Modelle mit totaler Variation
- III. Finite Elemente Methoden für die Minimierung der totalen Variation

Ein weiterer Aspekt dieser Arbeit, der sich über alle Teile erstreckt, ist die Entwicklung eines neuen Modells. Wir werden dieses theoretisch analysieren und entsprechende Lösungsmethoden entwickeln. Jeder Teil dieser Arbeit beginnt mit einer kurzen Einführung in den jeweiligen Themenbereich. Diese Einführungen enthalten eine Motivation in das entsprechende Forschungsgebiet, den Stand der Forschung mit Literaturüberblick und einen Überblick über den Beitrag, den diese Arbeit in dem jeweiligen Gebiet leistet. Die darauffolgenden Kapiteln bestehen jeweils aus Inhalten vom Autor veröffentlichter wissenschaftlicher Artikel. Das Layout der Artikel wurde vereinheitlicht. Eine inhaltliche Anpassung sowie Notationsanpassung erfolgt aber nicht, um den eigenständigen Charakter jeder Veröffentlichung beizubehalten.

Die Zusammenfassung ist, wie die gesamte Arbeit, in die oben genannten Teile gegliedert. Der Autor gibt in jedem der drei Teile einen Überblick über die Kapitel, wobei er die wesentlichen Inhalte, Errungenschaften und verwendeten Techniken wiedergibt.

## Unterraum-Korrekturverfahren für die Minimierung der totalen Variation

In Anwendungen der Bildverarbeitung (z.B. Entrauschen, Entzerren) ist man daran interessiert, aus gegebenen gestörten Daten das ursprüngliche Bild zu rekonstruieren. Dies ist im Allgemeinen ein schlecht-gestelltes inverses Problem, weshalb eine Rekonstruktion durch die Minimierung eines Funktionalen, bestehend aus einem Datenterm, der die Übereinstimmung zwischen Rekonstruktion und Beobachtung angibt, und einem Regularisierungsterm, der eine Überanpassung an die Beobachtung verhindert, bestimmt werden kann. Die Wahl des Regularisierungsterms hängt von den Eigenschaften (z.B. Glattheit) ab, die man von der Lösung fordert. Übliche Regularisierungsansätze, wie zum Beispiel die Tikhonov-Regularisierung, führen auf die Minimierung einer differenzierbaren Zielfunktion und somit auf eine glatte Lösung des inversen Problems. Allerdings ist man in der Bildverarbeitung daran interessiert, Kanten und Ecken in Bildern zu erhalten, weshalb die Lösung Unstetigkeiten erlauben muss und deshalb nicht notwendigerweise differenzierbar ist. Dies führt auf die Verwendung der *totalen*

*Variation* (TV) als Regularisierer. Das resultierende Optimierungsproblem erlaubt dann einerseits Unstetigkeiten und Sprünge in der Lösung, aber andererseits ist das Problem nicht mehr differenzierbar. Während die Entwicklung effizienter Methoden für glatte Optimierungsprobleme weitgehend gut verstanden ist, ist über die Konstruktion effizienter Methoden für die Minimierung der totalen Variation nur sehr wenig bekannt. Dies liegt an der Nicht-Additivität und Nicht-Glattheit der resultierenden TV-Minimierungsprobleme, was die Konstruktion solcher Methoden schwieriger macht. Im ersten Teil dieser Arbeit entwickeln wir einerseits genau solche effizienten Methoden für nicht-glatte und nicht-additive Variationsprobleme, wie der Minimierung der totalen Variation, mittels Unterraum-Korrekturverfahren und andererseits eine rigorose theoretische Konvergenzanalyse dieser neuen Methoden. Dabei konstruieren wir die ersten Gebietszerlegungsverfahren für die Minimierung der totalen Variation, deren Konvergenz zu einem Optimum des globalen Problems im Unendlichdimensionalen analytisch gesichert ist.

Effiziente Methoden sind genau dann besonders wichtig, wenn große Datenmengen verarbeitet werden müssen. Wir bemerken, dass durch die ständige Verbesserung der Hardware die Dimensionalität von Bildern und gemessenen Daten im Allgemeinen ständig steigt. Die so erhaltenen großen Datenmengen enthalten oft Rauschen und/oder andere Arten von Störungen, weshalb diese weiter bearbeitet werden müssen. Um diese großen Probleme in vernünftiger Zeit zu bewältigen, sind Unterraum-Korrekturverfahren und Gebietszerlegungsverfahren nötig, weil sie erlauben den Arbeitsaufwand zu zerteilen, indem anstatt eines Problems eine Folge von kleineren Problemen gelöst wird. Zusätzlich erlaubt diese Zerlegung, den Arbeitsaufwand auf mehrere Prozessorkerne aufzuteilen mit der Möglichkeit zur Parallelisierung.

In Kapitel 2, dessen Resultate in [AL1] veröffentlicht sind, konstruieren wir Unterraum-Korrekturverfahren für das Problem der gleichzeitigen Entfernung von Gaußschem Rauschen und Impulsrauschen aus gegebenen Bilddaten. Um so eine Art von gemischem Rauschen zu entfernen, entwickeln wir zuerst ein neues Modell beziehungsweise Optimierungsproblem, dessen Zielfunktion aus der totalen Variation und einer Kombination von einem quadratischen  $L^2$  Term und einem nicht-smoothen  $L^1$  Term besteht, d.h. für  $\Omega \subset \mathbb{R}^2$  betrachten wir

$$\min_{u \in L^2(\Omega)} \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \frac{\alpha_2}{2} \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \int_{\Omega} |Du|, \quad (0.1)$$

wobei, für  $i = 1, 2$ ,  $T_i : L^2(\Omega) \rightarrow L^2(\Omega)$  ein beschränkter linearer Operator ist,  $g_i \in L^2(\Omega)$  die gegebenen Daten sind,  $\alpha_i \geq 0$  mit  $\alpha_1 + \alpha_2 > 0$  und  $\int_{\Omega} |Du|$  die totale Variation von  $u$  in  $\Omega$  bezeichnet. Entsprechend nennen wir unser neues Modell  *$L^1$ - $L^2$ -TV Modell*. Um gemischtes Gauß-Impulsrauschen aus einem gegebenen Bild zu entfernen, setzt man typischerweise  $g_1 = g_2 =: g$  und  $T_1 = T_2 =: T$  im Modell (0.1). Wir demonstrieren mittels numerischer Experimente, dass der kombinierte Datenterm im Modell (0.1) für die Entfernung von derart gemischem Rauschen gut geeignet ist und visuell bessere Ergebnisse liefert als herkömmliche Verfahren. Man beachte, dass das Modell (0.1) eine Verallgemeinerung von bekannten und häufig verwendeten Modellen ist. Insbesondere für  $g_1 = g_2 =: g$  und  $T_1 = T_2 =: T$  erhalten wir mit  $\alpha_1 = 0$  für das Modell (0.1)

$$\min_{u \in L^2(\Omega)} \frac{\alpha_2}{2} \|Tu - g\|_{L^2(\Omega)}^2 + \int_{\Omega} |Du|, \quad (0.2)$$

welches wir  *$L^2$ -TV Modell* nennen, und für  $\alpha_2 = 0$  im Modell (0.1) erhalten wir das  *$L^1$ -TV Modell*

$$\min_{u \in L^2(\Omega)} \alpha_1 \|Tu - g\|_{L^1(\Omega)} + \int_{\Omega} |Du|. \quad (0.3)$$

Analytisch zeigen wir anhand eines konkreten Beispiels, dass die Minimierung des neu vorgestellten Funktionals bemerkenswerte Vorteile gegenüber den bekannten  $L^2$ -TV und  $L^1$ -TV Modellen hat. Das neue Modell übernimmt die vorteilhaften Eigenschaften dieser beiden Modelle und bessert unerwünschte Eigenschaften dieser aus. Zum Beispiel ist das  $L^2$ -TV Modell für  $T = I$  (Identität) strikt konvex, wobei das  $L^1$ -TV Modell nur konvex ist. Hingegen ist unser neues Modell wegen des kombinierten Datenterms wieder wünschenswerterweise strikt konvex.

Um das  $L^1$ - $L^2$ -TV Modell algorithmisch zu lösen, adaptieren wir einen Ansatz, der ursprünglich für das Lösen des reinen  $L^1$ -TV Modells entwickelt wurde. Allerdings kann diese numerische Methode für großdimensionale Bilder keine Lösung in absehbarer Zeit berechnen. Deshalb führen wir in [AL1] sequentielle und parallele sowie überlappende und nicht-überlappende Unterraum-Korrekturverfahren für die Minimierung des folgenden, etwas allgemeineren, nicht-glatten und nicht-additiven Problems ein

$$\min_{u \in L^2(\Omega)} \left\{ J(u) := \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \frac{\alpha_2}{2} \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \int_{\Omega} \varphi(|Du|) \right\}, \quad (0.4)$$

wobei  $\varphi(|\cdot|)$  eine konvexe Funktion von Maßen ist, welche den Regularisierer darstellt. Zum Beispiel erhalten wir für  $\varphi(x) = x$  das Problem (0.1). Entsprechend sind auch das  $L^2$ -TV Modell (0.2) und das  $L^1$ -TV Modell (0.3) konkrete Spezialfälle des Problems (0.4).

Für den konkreten Fall einer Zerlegung des Funktionenraums  $L^2(\Omega)$  in zwei Unterräume  $V_1$  und  $V_2$ , so dass  $L^2(\Omega) = V_1 + V_2$ , erhalten wir für Problem (0.4) folgendes paralleles Unterraum-Korrekturverfahren:

Initialisiere  $u^{(0)} =: \tilde{u}_1^{(0)} + \tilde{u}_2^{(0)} \in V_1 + V_2$  und iteriere

$$\begin{cases} u_1^{(n+1)} \leftarrow \arg \min_{u_1 \in V_1} J(u_1 + \tilde{u}_2^{(n)}), \\ u_2^{(n+1)} \leftarrow \arg \min_{u_2 \in V_2} J(\tilde{u}_1^{(n)} + u_2), \\ u^{(n+1)} \leftarrow \frac{u_1^{(n+1)} + u_2^{(n+1)} + u^{(n)}}{2}, \\ \tilde{u}_1^{(n+1)} \leftarrow \chi_1 \cdot u^{(n+1)}, \\ \tilde{u}_2^{(n+1)} \leftarrow \chi_2 \cdot u^{(n+1)}, \end{cases} \quad (0.5)$$

wobei  $\chi_1, \chi_2 \in L^\infty(\Omega)$  so, dass (a)  $\chi_1 + \chi_2 = 1$  und (b)  $\chi_i \in V_i$  für  $i = 1, 2$ . Die Mittelung im Update von  $u^{(n+1)}$  ist nötig, um die Konvergenz des parallelen Algorithmus zu gewährleisten. Auf ähnliche Weise konstruieren wir in [AL1] auch einen konvergenten sequentiellen Algorithmus, wobei hier keine Mittelung beim Update von  $u^{(n+1)}$  nötig ist, sondern  $u^{(n+1)}$  die Summe der Minimierer der Unterraumprobleme ist. Während die Konvergenz der vom entsprechenden Unterraum-Korrekturverfahren generierten Folge  $(u^{(n)})_{n \in \mathbb{N}}$  gesichert ist, können wir analytisch nicht garantieren, dass der Grenzwert von  $(u^{(n)})_{n \in \mathbb{N}}$  eine Lösung des ursprünglichen globalen Problems ist. Wir können aber eine Abschätzung zwischen der Entfernung einer Lösung des globalen Problems (0.4) und dem Grenzwert der Folge, die durch das Unterraum-Korrekturverfahren generiert wird, angeben. Diese Abschätzung ist in all unseren Experimenten tatsächlich immer sehr klein oder strebt sogar gegen Null, was darauf hindeutet, dass Konvergenz zum globalen Optimum erreicht wird. Weiter besitzen unsere Unterraum-Korrekturverfahren die Eigenschaft, dass die Energie  $J$  mit den Iterationen monoton abnimmt, d.h.  $J(u^{(0)}) \geq J(u^{(n)})$  für alle  $n \in \mathbb{N}$ .

Die Unterraum-Korrekturverfahren in [AL1] verlangen, dass die Optimierungsprobleme auf den Teilgebieten exakt gelöst werden; siehe (0.5). Dies ist in der Praxis nicht immer mög-

lich. Deshalb beschreiben wir in [AL2], dessen Inhalt in Kapitel 3 enthalten ist, Unterraum-Korrekturverfahren für das Problem (0.4), welche in jedem Teilgebiet eine Folge leichter handhabbarer Hilfsfunktionale minimieren. Diese Hilfsfunktionale sind bei einer Zerlegung in zwei Unterräume (wie oben) folgendermaßen definiert: Für  $a, u_i \in V_i$ ,  $u_{i^c} \in V_{i^c}$ ,  $i = 1, 2$  und  $i^c \in \{1, 2\} \setminus \{i\}$  ist

$$J^s(u_i + u_{i^c}, a + u_{i^c}) := J(u_i + u_{i^c}) + \frac{\alpha_2}{2} \left( \delta \|u_i - a\|_{L^2(\Omega)}^2 - \|T_2(u_i - a)\|_{L^2(\Omega)}^2 \right),$$

wobei  $\delta > \|T\|^2$ . Da die Unterraumprobleme unserer Methoden von der Form

$$\min_{u_i \in V_i} J(u_1 + u_2) \quad \text{für } i = 1, 2 \tag{0.6}$$

sind (z.B. im Fall von Algorithmus (0.5) ist  $u_{i^c} = \tilde{u}_{i^c}^{(n)}$  mit  $i^c \in \{1, 2\} \setminus \{i\}$ ), erhalten wir eine Näherungslösung von (0.6) mit Hilfe des folgenden iterativen Algorithmus: Für  $u_i^{(0)} \in V_i$ ,

$$u_i^{(\ell+1)} = \arg \min_{u_i \in V_i} J^s(u_i + u_{i^c}, u_i^{(\ell)} + u_{i^c}), \quad \ell \geq 0,$$

wobei  $u_{i^c} \in V_{i^c}$  und  $i^c \in \{1, 2\} \setminus \{i\}$ . Man beachte, dass diese Hilfsfunktionale für lineare und beschränkte Operatoren  $T_1$  und  $T_2$  strikt konvex sind. Für das resultierende Unterraum-Korrekturverfahren ist es in ähnlicher Weise wie in [AL1] möglich, neben der Konvergenz und Monotonie der Energie eine Abschätzung zwischen der berechneten Lösung und einem Minimierer des globalen Problems zu erhalten. Diese Abschätzung erlaubt uns numerisch zu zeigen, dass die von uns entwickelten Verfahren tatsächlich gegen den Minimierer des ursprünglichen globalen Problems konvergieren.

Anstatt ein Gebietszerlegungsverfahren für die Minimierung der nicht-glatten und nicht-additiven totalen Variation zu konstruieren, betrachten wir in Kapitel 4, dessen Ergebnisse in [AL3] veröffentlicht sind, das preduale Problem des  $L^2$ -TV Modells. Für  $T = I$  lautet das preduale Problem des  $L^2$ -TV Modells

$$\min \frac{1}{2} \|\operatorname{div} \mathbf{p} + g\|_{L^2(\Omega)}^2 \quad \text{über } \mathbf{p} \in H_0(\operatorname{div}, \Omega) \tag{0.7}$$

unter der Nebenbedingung (u.d.N.)  $-\lambda \mathbf{1} \leq \mathbf{p}(x) \leq \lambda \mathbf{1}$  für fast alle (f.f.a.)  $x \in \Omega$ ,

wobei  $H_0(\operatorname{div}, \Omega) := \{\mathbf{v} \in (L^2(\Omega))^2 : \operatorname{div} \mathbf{v} \in L^2(\Omega), \mathbf{v} \cdot \mathbf{n} = 0 \text{ auf } \partial\Omega\}$ ,  $\mathbf{n}$  die nach außen gerichtete Einheitsnormale auf  $\partial\Omega$  ist,  $\lambda = \frac{1}{\alpha_2}$  und  $\mathbf{1}$  den Einheitsvektor bezeichnet. Man beachte, dass die Zielfunktion des Problems (0.7) konvex, aber weder strikt konvex noch stark konvex ist, weshalb bekannte Ansätze zur Konstruktion von Unterraum-Korrekturverfahren nicht auf das Problem (0.7) anwendbar sind. Nichtsdestotrotz, wegen der glatten Zielfunktion und der punktweisen Nebenbedingung, erweist sich das Problem (0.7) geeigneter für eine Gebietszerlegung als die Struktur des  $L^2$ -TV Modells.

Dadurch motiviert, entwickeln wir in [AL3] sequentielle und parallele nicht-überlappende Gebietszerlegungsverfahren für (0.7) und zeigen analytisch im diskreten Fall, dass die Verfahren sogar zu einer Lösung des globalen Problems konvergieren. Numerische Experimente demonstrieren, dass dieser duale Gebietszerlegungsansatz schneller gegen eine Lösung konvergiert als der primale Ansatz von [AL1]. In diesem Sinne ist der Algorithmus in [AL3] dem in [AL1] nicht nur aus analytischer Sicht sondern auch aus praktischer Sicht überlegen. Auch wenn die theoretische Analyse in [AL3] nur für den Fall  $T = I$  durchgeführt ist, kann diese

leicht auf allgemeinere Fälle erweitert werden, zum Beispiel wenn  $T$  ein Faltungsoperator ist, wie [AL3, Remark 2] zeigt.

In Kapitel 5, welches die Ergebnisse von [AL4] enthält, beschäftigen wir uns wieder mit der Frage, wie wir doch noch einen Gebietszerlegungsalgorithmus für die direkte Minimierung der totalen Variation entwickeln können. Dabei gehen wir wie folgt vor. Wir betrachten ein Gebietszerlegungsverfahren für das preduale  $L^2$ -TV Modell (0.7). So ein Verfahren ist durch seine Teilprobleme und die entsprechenden Updates bestimmt. Durch Dualisierung dieser Teilprobleme erhalten wir neue Teilprobleme die im primalen Raum gegeben sind. Dies definiert uns dann ein Gebietszerlegungsverfahren für das  $L^2$ -TV Modell und nicht mehr seines predualen Problems. Die Wahl der Updates ergibt sich dann auch durch die des predualen Verfahrens. Diese Idee erlaubt uns erstmals, für die Minimierung der totalen Variation im Unendlichdimensionalen zur globalen Lösung konvergente parallele und sequentielle überlappende Gebietszerlegungsalgorithmen zu konstruieren. Man beachte, dass durch den Zusammenhang zum predualen Problem die Konvergenz zu einer Lösung des globalen Problems analytisch gesichert ist. Es sei an dieser Stelle erwähnt, dass dieser duale Zusammenhang nicht trivial ist. Dies liegt daran, dass die Teilprobleme aus keiner üblichen totalen Variation sondern einer lokal gewichteten totalen Variation bestehen, d.h. die Teilprobleme sind von der Gestalt

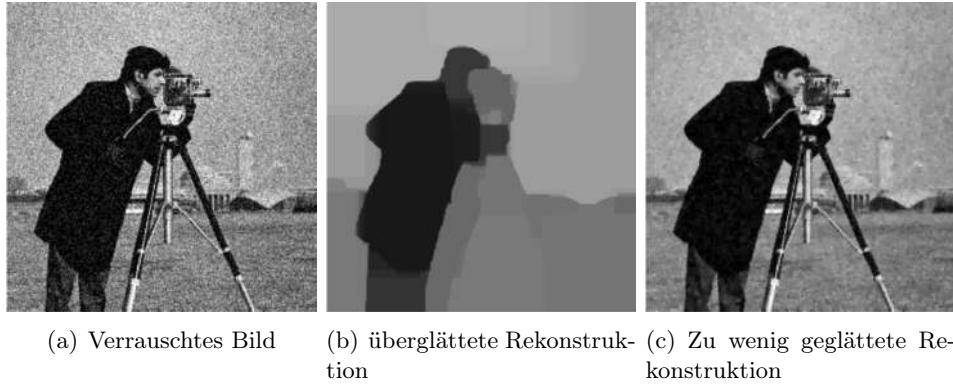
$$\min_{u \in L^2(\Omega)} \|u - f\|_{L^2(\Omega)}^2 + \int_{\Omega} \lambda |Du|,$$

wobei  $\lambda : \bar{\Omega} \rightarrow \mathbb{R}_0^+$  stetig und  $f \in L^2(\Omega)$  sind. Wir zeigen sowohl die Existenz und Eindeutigkeit der Lösung dieses lokal gewichteten TV-Problems als auch dessen Zusammenhang zu seinem predualen Problem. Erst dies erlaubt uns, die Konvergenz der überlappenden Gebietszerlegungsalgorithmen zu beweisen. Um die entwickelten Verfahren und die Lösung deren Teilprobleme praktisch umzusetzen, stellen wir zwei unterschiedliche Ansätze vor. Der erste Ansatz diskretisiert die Teilprobleme zuerst und optimiert anschließend, während der zweite Ansatz zuerst optimiert und erst dann diskretisiert. Der markanteste Unterschied dieser zwei Strategien ist, dass der erste Ansatz sogar eine nicht-überlappende Gebietszerlegung erlaubt, wobei dies der andere Ansatz analytisch nicht zulässt. Davon abgesehen, d.h. bei überlappenden Gebietszerlegungen, scheinen sich beide Ansätze in der Praxis fast gleich zu verhalten. Wir zeigen numerische Experimente, welche die theoretischen Resultate unterstützen und die Effektivität der neu entwickelten Verfahren demonstrieren.

## Parameterwahlmethoden für Modelle mit totaler Variation

Der zweite Teil dieser Arbeit beschäftigt sich mit Parameterwahlmethoden für die TV-Modelle (0.1), (0.2), und (0.3). Es ist offensichtlich, dass die Lösung dieser Optimierungsprobleme von der Wahl der Parameter  $\alpha_1$  und  $\alpha_2$  abhängt. Dabei ist die geeignete Wahl extrem wichtig; siehe Abbildung 0.1. Insbesondere führen zu klein gewählte Parameter zu einer überglätteten Rekonstruktion, womit nicht nur das Rauschen sondern auch die Details in den Bildern eliminiert werden. Setzt man hingegen  $\alpha_1$  und  $\alpha_2$  zu groß, so bleiben zwar einerseits die Details und kleinen Objekte der Bilder erhalten aber andererseits wird das Rauschen in homogenen Regionen nicht entfernt. Damit wir also eine gute Rekonstruktion erhalten, müssen die Parameter so gewählt werden, dass sie einen guten Kompromiss zwischen den vorhin genannten Eigenschaften bieten. Die bis jetzt üblichste Methode, Parameter in Anwendungen der Bildverarbeitung zu wählen, ist die sogenannte *Versuchs-und-Irrtums-Methode*, d.h. die Parameter werden durch

praktisches Herumprobieren bestimmt. Dies ist in der Regel entweder sehr aufwendig, weil mit einer sehr großen Anzahl an Parametern das betrachtete Problem gelöst werden muss, oder ungenau, wenn nur mit wenigen Parametern getestet wird. Wir entwickeln in dieser Arbeit neue Methoden, welche völlig automatisch und effizient Parameter bestimmen, die einen guten Kompromiss zwischen den oben beschriebenen Eigenschaften liefern.



**Abbildung 0.1:** Rekonstruktion eines Bildes, das Gaußsches Rauschen enthält, mittels (0.2) mit einem “relativ” kleinen Parameter  $\alpha_2$  in (b) und einem “relativ” großen Parameter  $\alpha_2$  in (c).

In Kapitel 7, welches die Ergebnisse von [AL5] beinhaltet, entwickeln wir automatische Parameterwahlmethoden für das  $L^1$ -TV Modell und das  $L^2$ -TV Modell, welche wir zusammenfassend in der Form

$$\min_{u \in L^2(\Omega)} \|Tu - g\|_{L^\tau(\Omega)}^\tau + \lambda \int_\Omega |Du|, \quad (0.8)$$

wobei  $\lambda > 0$  und  $\tau = 1, 2$  sind, schreiben. Analog könnte man statt (0.8) auch (0.2) und (0.3) betrachten. Um den Parameter  $\lambda$  geeignet zu wählen, wenden wir das Diskrepanzprinzip an. Dazu formulieren wir unser Bildverarbeitungsproblem als restringiertes Optimierungsproblem in der Form

$$\min_{u \in L^2(\Omega)} \int_\Omega |Du| \quad \text{u.d.N.} \quad \|Tu - g\|_{L^\tau(\Omega)}^\tau \leq \nu_\tau |\Omega|, \quad (0.9)$$

wobei  $\nu_\tau > 0$  eine Konstante ist, die vom Rauschen in den Daten abhängt,  $\tau = 1, 2$ , und  $|\Omega|$  das Volumen von  $\Omega$  bezeichnet. Das Diskrepanzprinzip besagt dann, dass der Parameter  $\lambda$  in (0.8) so zu bestimmen ist, dass die zugehörige Lösung  $u_\lambda$  von (0.8) die Nebenbedingung in (0.9) erfüllt. Um den Wert  $\nu_\tau$  zu bestimmen, ist Kenntnis über das Rauschniveau nötig. Dies mag bedeuten, dass womöglich in einem ersten Schritt das Rauschen geschätzt wird, bevor das Diskrepanzprinzip angewendet werden kann. Allerdings ist es im Allgemeinen leichter das Rauschen als den Regularisierungspараметer zu schätzen. Beachte, dass das restringierte Optimierungsproblem (0.9) auf natürliche Weise mit dem unrestringierten Problem (0.8) verbunden ist; siehe [AL5, Theorem 2.2]. Dieser Zusammenhang besteht darin, dass Parameter  $\lambda \geq 0$  existieren, so dass die Lösung von (0.8) der Lösung von (0.9) entspricht. In diesem Sinn sind diese Optimierungsprobleme äquivalent.

Basierend auf der Formulierung (0.9) und dem Diskrepanzprinzip präsentieren wir in [AL5] neue Methoden, welche automatisch den Parameter  $\lambda$  in (0.8) bestimmen. Unser Ansatz ist motiviert durch die iterative Parameterwahlmethode von [Chambolle, A.: An Algorithm for

Total Variation Minimization and Applications. *Journal of Mathematical Imaging and Vision* 20: 89–97, 2004], welche dort nur für das  $L^2$ -TV Modell mit  $T = I$  entwickelt wurde. Wir haben in [AL5] diesen Algorithmus für das  $L^2$ -TV Modell und das  $L^1$ -TV Modell für allgemeine lineare und beschränkte Operatoren  $T$  adaptiert. Beachte, dass es sich dabei nicht um eine einfache Erweiterung des Algorithmus von Chambolle auf allgemeinere Situationen handelt. Dies liegt darin, dass in jeder Iteration der vorläufig geschätzte Parameter upgedatet wird. Dieses Update für  $\lambda$  muss so gewählt sein, dass das Verfahren gegen den gewünschten Regularisierungsparameter konvergiert. Dazu müssen wir in unserem allgemeinen Fall ein geeignetes Update wählen, welches sich von dem aus der Arbeit von Chambolle deutlich unterscheidet. Dies gewährleistet, dass unsere Methode theoretisch und auch numerisch gegen den gewünschten Parameter konvergiert. Neben der allgemeinen Anwendbarkeit unserer Parameterwahlmethode, besitzt sie sogar Vorteile im Fall  $\tau = 2$  und  $T = I$  gegenüber dem Algorithmus von Chambolle. Im Besonderen gilt, dass in diesem Fall unsere Methode aus [AL5] höchstens so viele Iterationen benötigt wie die Methode von Chambolle und in der Praxis häufig weniger. Dieses Verhalten ist nicht nur numerisch sondern auch theoretisch erklärbar [AL5, Abschnitt 6.1].

Wir bemerken, dass ein skalarer Regularisierungsparameter nicht notwendigerweise die beste Wahl für jedes Bildrekonstruktionsproblem ist, weil Bilder üblicherweise große homogene Regionen sowie auch Teile mit vielen Details besitzen. Es scheint also besser zu sein, wenn  $\lambda$  in homogenen Regionen groß ist, um dort das Rauschen deutlich zu reduzieren, und in den Regionen mit Details relativ klein ist, um diese zu erhalten. Mit dieser Absicht betrachten wir in [AL5] auch das Problem

$$\min_{u \in L^2(\Omega)} \|Tu - g\|_{L^\tau(\Omega)}^\tau + \int_\Omega \lambda(x)|Du| \quad (0.10)$$

für  $\tau = 1, 2$ , wobei  $\lambda : \Omega \rightarrow \mathbb{R}^+$  eine beschränkte und stetige Funktion ist. Um  $\lambda$  entsprechend lokal adaptiv zu wählen, so dass Details erhalten bleiben und Rauschen in uniformen Teilen merklich eliminiert wird, konstruieren wir in [AL5] einen iterativen Algorithmus basierend auf lokalen Nebenbedingungen. Das heißt anstelle von (0.9) betrachten wir für  $\tau = 1, 2$

$$\min_{u \in L^2(\Omega)} \int_\Omega |Du| \quad \text{u.d.N.} \quad S(u)(x) := \int_\Omega w(x, y)|Tu - g|^\tau(y)dy \leq \nu_\tau \quad \text{f.f.a.} \quad x \in \Omega, \quad (0.11)$$

wobei  $w$  ein normalisierter Filter ist, d.h.

$$w \in L^\infty(\Omega \times \Omega), \quad w \geq 0 \text{ auf } \Omega \times \Omega, \quad \int_\Omega \int_\Omega w(x, y)dydx = 1 \quad \text{und} \quad (0.12)$$

$$\exists \epsilon > 0 : \int_\Omega \int_\Omega w(x, y)|\phi(y)|^\tau dydx \geq \epsilon \|\phi\|_{L^\tau(\Omega)}^\tau \quad \forall \phi \in L^\tau(\Omega), \quad (0.13)$$

wobei  $\epsilon$  nicht von  $\phi$  abhängt. Wählen wir  $w$  so, dass die Ungleichung (0.13) eine Gleichung mit  $\epsilon = \frac{1}{|\Omega|}$  wird, dann zeigen wir in [AL5, Proposition 4.2], dass für die Lösung  $u_l$  des lokal restriktionsierten Minimierungsproblems (0.11)

$$\|Tu_l - g\|_{L^\tau(\Omega)}^\tau \leq \nu_\tau |\Omega| \quad \text{und} \quad \int_\Omega |Du_l| \geq \int_\Omega |Du_g|$$

gilt, wobei  $u_g$  eine Lösung von (0.9) ist. Wir sehen also, dass die totale Variation von  $u_l$  größer gleich der von  $u_g$  ist. Folglich können wir erwarten, dass die Lösung des lokal restriktionsierten

Problems Details und Kanten besser als der Minimierer des global restriktierten Problems bewahrt. Um eine geeignete Wahl des Parameters  $\lambda$  in (0.10) unter Verwendung von (0.11) zu erhalten, verfahren wir in [AL5] wie folgt: Sei  $\lambda$  gegeben und  $u_\lambda$  die entsprechende Lösung von (0.10). Ist nun  $S(u_\lambda)(x) > \nu_\tau$ , so nehmen wir an, dass Bilddetails im lokalen Residuum enthalten sind und deshalb verringern wir  $\lambda$  in  $x$ . Im Fall  $S(u_\lambda)(x) < \nu_\tau$  erhöhen wir  $\lambda$  in  $x$  und falls  $S(u_\lambda)(x) = \nu_\tau$  bleibt  $\lambda$  in  $x$  unverändert. Mit Hilfe dieser Vorschrift konstruieren wir Algorithmen, welche automatisch lokal adaptive  $\lambda$  erzeugen. Numerische Beispiele zeigen, dass die so erhaltenen  $\lambda$  Lösungen  $u_\lambda$  generieren, welche qualitativ besser sind als die Rekonstruktionen, die mittels global konstanten Parametern erzeugt werden.

Anstelle von (0.10) kann man auch das Optimierungsproblem

$$\min_{u \in L^2(\Omega)} \int_{\Omega} \alpha(x) |Tu - g|^\tau dx + \int_{\Omega} |Du|, \quad (0.14)$$

wobei  $\alpha : \Omega \rightarrow \mathbb{R}^+$ , betrachten. Man beachte, dass für skalare Parameter  $\alpha, \lambda > 0$  die Probleme (0.10) und (0.14) mit  $\alpha = \frac{1}{\lambda}$  äquivalent sind. Man könnte für (0.14) in ähnlicher Weise wie vorhin eine Parameterwahlmethode konstruieren. Allerdings würde dies nur sinnvoll sein, wenn die Ausgabe von  $T$  wiederum über den Bildbereich definiert ist, weil sonst nicht klar ist, wie  $\alpha$  lokal zu wählen ist. Ist zum Beispiel  $T$  eine Abbildung vom Bildbereich in einen Frequenzraum, so ist unklar in welchen Frequenzbereichen  $\alpha$  groß oder klein zu wählen ist, um die Rekonstruktion zu verbessern. Dies begrenzt die Anwendbarkeit dieses Ansatzes für Anwendungen in denen  $T$  eine Transformation eines Bildes in eine andere Art von Datenraum ist. Beispiele solcher Transformationen beinhalten die Wavelet- und Fourier-Transformation. In so einem Fall ändert sich (0.14) für  $\tau = 2$  zu

$$\min_{u \in L^2(\Omega)} \int_{\Lambda} \alpha(x) |Tu - g|^2 dx + \int_{\Omega} |Du|, \quad (0.15)$$

wobei  $\Lambda$  ein entsprechender Datenbereich (z.B. Frequenzraum) ist. Beachte, dass für (0.10) diese Limitierung nicht zutrifft, weil  $\lambda$  im TV Term steht, welcher in diesen Anwendungen auch auf dem Bildbereich  $\Omega$  definiert ist; vergleiche mit (0.15). In [AL6, AL7], dessen Ergebnisse in Kapitel 8 enthalten sind, betrachten wir (0.15), wobei  $T$  entweder aus einer Wavelet-Transformation oder Fourier-Transformation besteht. Genauer gesagt, betrachten wir die Anwendung der Rekonstruktion eines Bildes aus unvollständigen Fourierkoeffizienten, wie es bei der medizinischen Bildverarbeitung vorkommt (MRT), oder verlorengegangene Waveletkoeffizienten, was zum Beispiel bei der Übertragung von komprimierten Bilddaten auftritt. Für diese Probleme konstruieren wir eine automatische lokal adaptive Parameterwahlmethode, indem wir anstelle von (0.15) eine Folge von Hilfsproblemen lösen. Diese Hilfsprobleme sind so gewählt, dass (i)  $T$  nicht mehr auf  $u$  wirkt und so jedes Hilfsproblem im Bildbereich gelöst werden kann und (ii) die Folge der Lösungen der Hilfsprobleme gegen eine Lösung des ursprünglichen Problems konvergiert. Da die Hilfsprobleme komplett im Bildbereich gestellt sind und dort auch gelöst werden, ist es uns in [AL6, AL7] möglich, lokal variierende Gewichte zu wählen. Numerische Experimente zeigen, dass mit diesem neuen Verfahren bei der Rekonstruktion von partiellen Fourierdaten und bei der Rekonstruktion von verlorengegangenen Waveletkoeffizienten bessere Resultate erzielt werden können, als von herkömmlichen Verfahren mit einem global konstanten Parameter.

Falls der dynamische Bereich  $[c_{\min}, c_{\max}]$  in dem das originale Bild  $\hat{u}$  liegt, d.h.  $c_{\min} \leq \hat{u}(x) \leq c_{\max}$  f.f.a.  $x \in \Omega$ , bekannt ist, scheint es sinnvoll, diese Information auch im Rekonstruktionsprozess beziehungsweise im Optimierungsproblem mitzuverwenden. Betrachten wir

das Optimierungsproblem (0.10) für  $\tau = 2$ , so würde dies dann

$$\min_{u \in L^2(\Omega) \cap C} \|Tu - g\|_{L^2(\Omega)}^2 + \int_{\Omega} \lambda(x)|Du| \quad (0.16)$$

ergeben, wobei  $C := \{u \in L^2(\Omega) : c_{\min} \leq u(x) \leq c_{\max} \text{ f.f.a. } x \in \Omega\}$  die Box-Nebenbedingung bezeichnet. Es ist klar, dass im Allgemeinen eine Lösung des  $L^2$ -TV Modells nicht in der Menge  $C$  liegt. Allerdings, wenn der Regularisierungsparameter  $\lambda$  im  $L^2$ -TV Modell, siehe (0.8), groß genug gewählt wird, dann liegen auch die entsprechenden Minimierer in  $C$ . In diesem Fall hat also die Box-Nebenbedingung in (0.16) keinen Einfluss auf die Lösung. Dies wirft die Frage auf, ob ein optimal gewählter Parameter eine Lösung ergibt, die sowieso in  $C$  liegt und wir deshalb diese Box-Nebenbedingung einfach weglassen können. In Kapitel 9, das die Resultate von [AL8] beinhaltet, befassen wir uns mit dieser Fragestellung. Dazu berechnen wir numerisch nahezu optimale konstante Parameter sowie lokal adaptive  $\lambda$  für das  $L^2$ -TV Modell und das restriktierte Problem (0.16). Ein Vergleich der daraus erhaltenen Rekonstruktionen zeigt, dass die Wahl des Parameters wichtiger ist als die Verwendung von Box-Nebenbedingung. Genauer gesagt unterscheiden sich die Rekonstruktionen von (0.8) (mit  $\tau = 2$ ) und (0.16) bei nahezu optimaler Parameterwahl so gut wie nicht in den meisten Anwendungen. Für bestimmte Fälle können wir sogar analytisch nachweisen, dass die Lösung des Optimierungsproblems ohne Box-Nebenbedingung tatsächlich in der Menge  $C$  liegt und so die Box-Nebenbedingung keinen Einfluss auf die Lösung hat. Allerdings bei der Rekonstruktion von abgetasteten Radon-Daten, wie sie bei der Computertomographie auftreten, können wir feststellen, dass die zusätzliche Nebenbedingung die Rekonstruktion entscheidend verbessert. Basierend auf der Beobachtung, dass in vielen Anwendungen die Box-Nebenbedingungen Einfluss auf die Lösung hat, würde es Sinn machen, eine Parameterwahlmethode basierend auf dem Bildbereich zu konstruieren. Dies ist genau dann von Nutzen, wenn das Rauschniveau nicht bekannt ist, hingegen der Bildbereich, in dem die Rekonstruktion liegen soll. Für diesen Fall konstruieren wir in [AL8] eine automatische Parameterwahlmethode.

Anstatt anzunehmen, dass ein beobachtetes Bild nur eine Art von Rauschen enthält, ist ein viel realistischerer Ansatz, dass eine Vermischung von verschiedenen Arten von Rauschen enthalten sind. In diesem Sinn wird in [AL1] das  $L^1$ - $L^2$ -TV Modell (0.1) entwickelt, um Gaußsches Rauschen und Impulsrauschen gleichzeitig zu eliminieren. Wie schon erwähnt und ähnlich wie im Fall der  $L^2$ -TV und  $L^1$ -TV Modelle, hängt die Lösung des  $L^1$ - $L^2$ -TV Modells entscheidend von der Wahl von  $\alpha_1$  und  $\alpha_2$  ab. Basierend auf einer stochastischen Charakterisierung des gemischten Gauß-Impulsrauschens, stellen wir in Kapitel 10, das die Ergebnisse aus [AL9] wiedergibt, eine vollständig automatische Parameterwahlstrategie zur Bestimmung von  $\alpha_1$  und  $\alpha_2$  im  $L^1$ - $L^2$ -TV Modell (0.1) vor. Diese Strategie basiert ähnlich wie die Verfahren in [AL5] auf dem Diskrepanzprinzip. Deshalb bringen wir das  $L^1$ - $L^2$ -TV Modell mit einem entsprechenden restriktierten Optimierungsproblem in Verbindung. Im Speziellen betrachten wir das restriktierte Problem

$$\min_{u \in L^2(\Omega)} \int_{\Omega} |Du| \quad \text{u.d.N.} \quad \|T_1 u - g_1\|_{L^1(\Omega)} \leq \nu_1 |\Omega| \quad \text{und} \quad \|T_2 u - g_2\|_{L^2(\Omega)}^2 \leq \nu_2 |\Omega|, \quad (0.17)$$

wobei  $\nu_1, \nu_2 \geq 0$  stochastische Werte abhängig vom Rauschen sind. Unter den Voraussetzungen, dass es ein  $u$  gibt, das die Nebenbedingungen in (0.17) erfüllt, und es ein  $i \in \{1, 2\}$  gibt, so dass konstante Funktionen nicht im Kern von  $T_i$  enthalten sind, besitzt (0.17) eine Lösung [AL9, Theorem 3.2]. Unter ähnlichen Voraussetzungen (ohne die Realisierbarkeitsbedingung

an die Nebenbedingungen in (0.17)) beweisen wir in [AL9] auch die Existenz eines Minimierers des  $L^1$ - $L^2$ -TV Modells. Darüberhinaus können wir zeigen, dass es  $\alpha_1$  und  $\alpha_2$  gibt, so dass die Lösung des unrestringierten  $L^1$ - $L^2$ -TV Modells auch Lösung des restringierten Problems (0.17) ist. Dieses theoretische Resultat erlaubt uns in [AL9] eine automatische Parameterwahlmethode für  $\alpha_1$  und  $\alpha_2$  zu konstruieren. Im Speziellen basiert diese Methode auf dem Diskrepanzprinzip und verwendet in jeder Iteration ein Zusammenspiel zwischen dem  $L^1$ - $L^2$ -TV Modell und dem entsprechenden restringierten Problem. Dies erlaubt dem Algorithmus automatisch zu erkennen ob der Parameter  $\alpha_i$ ,  $i = 1, 2$ , verkleinert oder vergrößert werden soll. Da diese Methode eine monotone Folge von Parametern generiert, können wir dessen Konvergenz zeigen. Numerische Experimente demonstrieren, dass die beschriebene Methode nahezu optimale Parameter  $\alpha_1$  und  $\alpha_2$  findet, so dass die dazugehörige Rekonstruktion visuell zufriedenstellend ist.

In jeder Iteration der Parameterwahlmethode müssen wir das  $L^1$ - $L^2$ -TV Modell mit den momentanen Parametern lösen. Dazu verwenden wir den Algorithmus aus [AL1] und beweisen in [AL9] dessen Konvergenz analytisch.

Da lokal adaptive Parameter bessere Rekonstruktionen ergeben können als skalare, betrachten wir in Kapitel 11, dessen Resultate in [AL10] veröffentlicht sind, ein lokal adaptives  $L^1$ - $L^2$ -TV Modell, nämlich

$$\min_{u \in L^2(\Omega)} \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \int_{\Omega} \lambda(x) |Du|. \quad (0.18)$$

Unter den gleichen Voraussetzungen wie in [AL9], ist die Existenz einer Lösung von (0.18) gesichert [AL10]. Um  $\lambda$  entsprechend berechnen zu können, betrachten wir wieder ein Optimierungsproblem der Form (0.11), wobei hier

$$S(u)(\cdot) := \int_{\Omega} w(\cdot, y) \alpha_1 |T_1 u - g_1|(y) + \alpha_2 |T_2 u - g_2|^2(y) dy$$

und  $w$  ein normalisierter Lokalisierungsfilter ist, für den (0.12) und (0.13) für  $\tau = 1, 2$  gilt. Zusätzlich setzen wir  $\nu_{\tau} = B$ , da die rechte Seite der Ungleichung in (0.11) jetzt nicht mehr von  $\tau$  abhängt. Wir verfolgen zwei Ansätze bezüglich der Wahl von  $B$ . Im ersten Ansatz wird  $B$  für jede lokale Nebenbedingung konstant gleich gesetzt, wobei im zweiten Ansatz  $B$  lokal verschieden gewählt wird. Basierend auf einem Algorithmus ähnlich zu dem in [AL5] zeigen wir in [AL10], dass ein lokal angepasstes  $B$  weit bessere Ergebnisse liefern kann als ein global konstantes  $B$ . Dies erlaubt uns die Stärke von lokal variierenden Parametern zu demonstrieren und es unterstreicht, wie wichtig geeignete gewählte lokale Nebenbedingungen sind.

## Finite Elemente Methoden für die Minimierung der totalen Variation

Während wir in den vorangegangenen Teilen bei der numerischen Implementierung eine Finite Differenzen Diskretisierung des entsprechenden TV Problems verwendet haben, beschäftigen wir uns im dritten Teil dieser Arbeit mit einer Finiten Elemente Diskretisierung der totalen Variation. Der Grund, weshalb hauptsächlich Finite Differenzen für Bildrekonstruktionsprobleme verwendet werden, mag sein, dass die Bildinformation pixelweise gespeichert wird und dass die resultierenden Optimierungsprobleme üblicherweise ein nicht-differenzierbares Zielfunktional besitzen, welches keine herkömmliche Ableitung erlaubt. Dies macht es relativ schwer, die

schwache Formulierung zu finden, welche für Finite Elemente Methoden benötigt wird. Aber es gibt gute Gründe, eine Finite Elemente Diskretisierung für Bildverarbeitungsprobleme und somit für die Minimierung der totalen Variation zu betrachten. Zum Beispiel benötigen wir eine Finite Elemente Diskretisierung, wenn die Bilddaten nicht auf einem kartesischen Gitter gegeben sind, was der Fall ist, wenn das Aufnahmegerät hexagonale Sensoren besitzt. Weiter erlauben Finite Elemente eine Bildrekonstruktion auf einer Mannigfaltigkeit und finden Anwendung in der medizinischen Bildverarbeitung. Man beachte, dass zur Lösung von partiellen Differentialgleichungen Finite Elemente Methoden weit verbreitet sind, weil sie wegen ihrer Flexibilität gut komplexe Geometrien und komplizierte Randbedingungen behandeln können. Ein noch größerer Vorteil ist, dass sie eine adaptive Diskretisierung relativ einfach erlauben. In diesem Teil der Arbeit, der den Inhalt der Publikation [AL11] beinhaltet, präsentieren wir eine effiziente Implementierung des  $L^1$ - $L^2$ -TV Modells mit  $T_1 = T_2 = I$  mittels einer Finiten Elemente Diskretisierung über konforme, lokal verfeinerte Gitter. Darauf hinaus zeigen wir, dass ein Minimierer des betrachteten Modells über einem geeigneten Finiten Elemente Raum gegen einen Minimierer des ursprünglichen Problems im Raum der Funktion mit beschränkter Variation konvergiert. Bei dem geeigneten Finiten Elemente Raum handelt es sich um den Raum der stückweise affinen und global stetigen Funktionen. Man beachte, dass diese  $\Gamma$ -Konvergenz im Allgemeinen nicht für alle Finiten Elemente Räume gilt, wie man mittels Gegenbeispiel zeigen kann.

Die effiziente Implementierung des betrachteten Optimierungsproblems basiert auf einem primal-dualen Algorithmus und der Programmierumgebung DUNE. Durch die Verwendung von DUNE-ALUGRID und durch die Fähigkeiten von DUNE-ACFEM ist der resultierende Algorithmus parallelisiert. Weiter basiert die Implementierung auf einer adaptiven Verfeinerung des Gitters, wobei heuristisch an bekannten Unstetigkeiten lokal verfeinert wird. Dies zeigt, dass eine effiziente Finite Elemente Implementierung für die Minimierung der (nicht-smooth) totalen Variation tatsächlich möglich ist.

## Literaturverzeichnis

- [AL1] M. Hintermüller und A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1$ / $L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.
- [AL2] M. Hintermüller und A. Langer. Surrogate functional based subspace correction methods for image processing. In *Domain Decomposition Methods in Science and Engineering XXI*, pages 829–837, Springer, 2014.
- [AL3] M. Hintermüller und A. Langer. Non-overlapping domain decomposition methods for dual total variation based image denoising. *Journal of Scientific Computing*, 62(2):456–481, 2015.
- [AL4] A. Langer. Overlapping domain decomposition methods for total variation denoising. *submitted to SIAM Journal on Numerical Analysis*, 2018.
- [AL5] A. Langer. Automated parameter selection for total variation minimization in image restoration. *Journal of Mathematical Imaging and Vision*, 57(2):230–268, 2017.
- [AL6] M. Hintermüller und A. Langer. Adaptive regularization for Parseval frames in image processing. *SFB-Report No. 2014-014*, 2014.

- [AL7] M. Hintermüller, A. Langer, C. N. Rautenberg und T. Wu. Adaptive regularization for image reconstruction from subsampled data. In *Proceedings of the International Conference on Imaging, Vision and Learning Based Optimization and PDEs (ILVOPDE)*, pages 3–26, Springer, 2016.
- [AL8] A. Langer. Investigating the influence of box-constraints on the solution of a total variation model via an efficient primal-dual method. *Journal of Imaging*, 4(1):12, 2018.
- [AL9] A. Langer. Automated parameter selection in the  $L^1$ - $L^2$ -TV model for removing Gaussian plus impulse noise. *Inverse Problems*, 33:41, 2017.
- [AL10] A. Langer. Locally adaptive total variation for removing mixed Gaussian–impulse noise. *International Journal of Computer Mathematics*, 2018, DOI: 10.1080/00207160.2018.1438603.
- [AL11] M. Alkämper und A. Langer. Using DUNE-ACFem for non-smooth minimization of bounded variation functions. *Archive of Numerical Software*, 5(1):3–19, 2017.

# Outline

In this habilitation thesis, new efficient numerical methods for minimizing the total variation and their theoretical and numerical analysis are presented. For developing such methods we concentrate on three different classes of approaches, namely subspace correction methods, parameter selection methods and finite element methods. In this spirit the thesis is divided into three parts:

I. Subspace Correction Methods for Total Variation Minimization

II. Parameter Selection Methods for Total Variation Models

III. Finite Element Discretization of the Total Variation

A further topic of this thesis, which spans over all parts, is the development of a new total variation model. This new model is theoretically analyzed and corresponding solution strategies are proposed. All three parts start with a brief introduction describing, apart from a short motivation, the stat-of-the-art and the contribution of the author in the respective research fields. The subsequent chapters contain own published scientific results. The layout of each publication is adjusted to the layout of the thesis, but the content and notations are kept unchanged to preserve the independent character of each publication.

## List of own publications contained in the thesis

I. Subspace Correction Methods for Total Variation Minimization

- [AL1] M. Hintermüller and A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1/L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.
- [AL2] M. Hintermüller and A. Langer. Surrogate functional based subspace correction methods for image processing. In Domain Decomposition Methods in Science and Engineering XXI, pages 829–837, Springer, 2014.
- [AL3] M. Hintermüller and A. Langer. Non-overlapping domain decomposition methods for dual total variation based image denoising. *Journal of Scientific Computing*, 62(2):456–481, 2015.
- [AL4] A. Langer. Overlapping domain decomposition methods for total variation denoising. *submitted to SIAM Journal on Numerical Analysis*, May 2018.

## II. Parameter Selection Methods for Total Variation Models

- [AL5] A. Langer. Automated parameter selection for total variation minimization in image restoration. *Journal of Mathematical Imaging and Vision*, 57(2):230–268, 2017.
- [AL6] M. Hintermüller and A. Langer. Adaptive regularization for Parseval frames in image processing. *SFB-Report No. 2014-014*, 2014.
- [AL7] M. Hintermüller, A. Langer, C. N. Rautenberg, and T. Wu. Adaptive regularization for image reconstruction from subsampled data. In *Proceedings of the International Conference on Imaging, Vision and Learning Based Optimization and PDEs (ILVOPDE)*, pages 3–26, Springer, 2016.
- [AL8] A. Langer. Investigating the influence of box-constraints on the solution of a total variation model via an efficient primal-dual method. *Journal of Imaging*, 4(1):12, 2018.
- [AL9] A. Langer. Automated parameter selection in the  $L^1$ - $L^2$ -TV model for removing Gaussian plus impulse noise. *Inverse Problems*, 33:41, 2017.
- [AL10] A. Langer. Locally adaptive total variation for removing mixed Gaussian-impulse noise. *International Journal of Computer Mathematics*, 2018, DOI: 10.1080/00207160.2018.1438603.

## III. Finite Element Discretization of the Total Variation

- [AL11] M. Alkämper and A. Langer. Using DUNE-ACFem for non-smooth minimization of bounded variation functions. *Archive of Numerical Software*, 5(1):3–19, 2017.

## Part I.

# Subspace Correction Methods for Total Variation Minimization



# 1. Introduction and Overview

It has turned out, that variational methods are successful solution strategies for solving problems in image reconstruction (denoising, deblurring, inpainting, super-resolution, etc.) and for problems in image analysis (e.g., segmentation and estimation of optical flow). In this thesis we restrict ourselves to applications in image reconstruction and refer to [4] for an introduction to variational approaches in image analysis.

In image reconstruction the obtained data  $g$  can be modelled as

$$g = \mathcal{N}(T\hat{u}),$$

where  $\hat{u}$  is the unknown original image,  $T$  is a linear bounded operator, modelling the dependency of data on the underlying image, and  $\mathcal{N}$  represents noise. Typical examples for  $T$  are (i) convolution operators, which describe blur in an image, (ii) the identity operator, if an image is only corrupted by noise, (iii) the characteristic function of a subdomain marking missing parts, i.e., the inpainting domain, or (iv) the Fourier-transform, if the observed data are given as corresponding frequencies. Different types of noise are usually modelled based on probability distributions. Prominent examples are Gaussian noise, impulse noise, Poisson noise, Rician noise, and multiplicative noise. In the sequel we restrict ourselves to Gaussian noise and impulse noise and refer the reader regarding multiplicative noise, Rician noise, and Poisson noise to [3, 27, 35] and references therein. Measured data which possess additive Gaussian noise only are described as

$$g = T\hat{u} + \eta,$$

where  $\eta$  represents the noise. In case of impulse noise, there are two models mainly used in the literature. The first one is called *salt-and-pepper noise*, where the noisy data  $g$  are given by

$$g(x) = \begin{cases} c_{\min} & \text{with probability } s_1 \in [0, 1), \\ c_{\max} & \text{with probability } s_2 \in [0, 1), \\ T\hat{u} & \text{with probability } 1 - s_1 - s_2 > 0, \end{cases}$$

with  $T\hat{u}$  being in the image intensity range  $[c_{\min}, c_{\max}]$ , i.e.,  $c_{\min} \leq T\hat{u} \leq c_{\max}$  [12]. The second model is named *random-valued impulse noise*, where  $g$  is described as

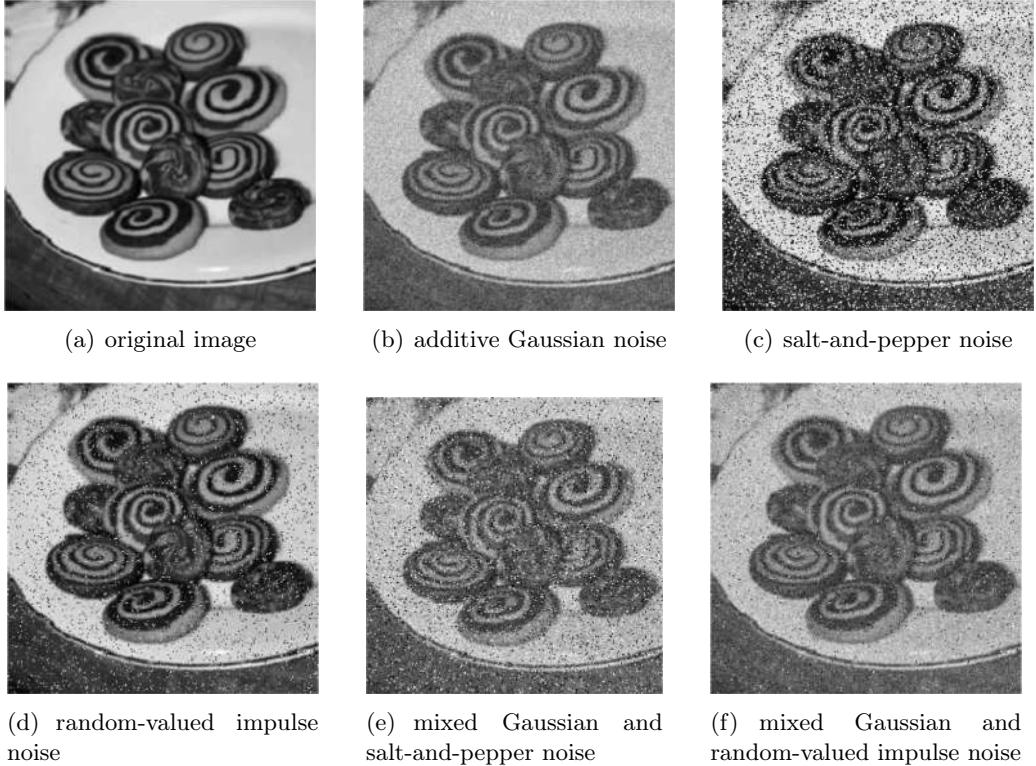
$$g(x) = \begin{cases} \rho & \text{with probability } s \in [0, 1), \\ T\hat{u} & \text{with probability } 1 - s. \end{cases}$$

Here  $\rho$  is a uniformly distributed random variable in the image intensity range  $[c_{\min}, c_{\max}]$ . In Figure 1.1 we show some typical examples of noisy images.

The reconstruction of  $\hat{u}$  from the given degraded data  $g$  is an ill-posed inverse problem and therefore regularization techniques are required to recover the unknown image [20]. A good approximation of  $\hat{u}$  may be obtained by solving a minimization problem of the type

$$\min_u \mathcal{H}(Tu; g) + \lambda \mathcal{R}(u),$$

## I. Subspace Correction Methods for Total Variation Minimization



**Figure 1.1:** Original image and its contamination by different types of noise.

where  $\mathcal{H}(\cdot; g)$  denotes the data-term, which enforces the consistency between the observed and reconstructed data,  $\mathcal{R}$  is an appropriate filter or regularization term preventing overfitting and  $\lambda > 0$  is a parameter weighting the influence of the two terms.

Since images usually contain edges and discontinuities, the *total variation* (TV), which is known to preserve these features, plays a fundamental role in imaging as a regularization technique. Starting with the pioneering work [49], where the total variation was used for denoising, total variation minimization has become more and more important in image reconstruction. We recall, that for  $u \in L^1(\Omega)$ , where  $\Omega \subset \mathbb{R}^d$ ,  $d \in \mathbb{N}$ , is an open bounded set with Lipschitz boundary,

$$V(u, \Omega) := \sup \left\{ \int_{\Omega} u \operatorname{div} \mathbf{p} \, dx : \mathbf{p} \in (C_c^1(\Omega))^d, \|\mathbf{p}\|_{\infty} \leq 1 \right\}$$

denotes the *variation* of  $u$  in  $\Omega$ . Here  $C_c^1(\Omega)$  is the space of  $C^1$ -functions with compact support in  $\Omega$  and  $\|\mathbf{p}\|_{\infty} = \sup_x \sqrt{\sum_i p_i^2(x)}$ , where  $\mathbf{p} = (p_1, \dots, p_d)$ . If  $V(u, \Omega) < \infty$ , then we denote  $\int_{\Omega} |Du| = V(u, \Omega)$  and call it the total variation of  $u$  in  $\Omega$  (see [2] for more details). Further, we denote by  $BV(\Omega)$  the space of functions with bounded variation, i.e.,  $u \in BV(\Omega)$  if and only if  $V(u, \Omega) < \infty$ . The space  $BV(\Omega)$  endowed with the norm  $\|u\|_{BV(\Omega)} = \|u\|_{L^1(\Omega)} + \int_{\Omega} |Du|$  is a nonreflexive Banach space [30]. Since the total variation preserves edges and discontinuities in images, we use in the sequel always  $\mathcal{R}(u) = \int_{\Omega} |Du|$ . However, we note that depending on the application other regularization terms, such as the total generalized variation [6], the nonlocal total variation [29, 34, 63], the Mumford-Shah regularizer [43], or higher order regularizer (see

## 1. Introduction and Overview

[46] and references therein) might be used as well.

The choice of the data-term  $\mathcal{H}$  usually depends on the type of noise contained in the image. In this vein one typically considers a quadratic  $L^2$  data-term, if the image is contaminated by Gaussian noise only, see for example [7, 10]. In this approach, which we call  *$L^2$ -TV model*, a reconstruction from the observed data  $g$  is obtained by solving

$$\min_{u \in BV(\Omega)} \|Tu - g\|_{L^2(\Omega)}^2 + \lambda \int_{\Omega} |Du|, \quad (1.1)$$

where  $T : L^2(\Omega) \rightarrow L^2(\Omega)$  is bounded and linear. There exists a variety of first order methods, see for example [7, 11] and references therein, and second order methods [32], which perform well for solving the optimization problem (1.1).

The  $L^2$ -TV model usually does not generate a satisfactory reconstruction if instead of Gaussian noise only impulse noise is present in an image. Impulse noise is constituted for example by malfunctioning pixels in camera sensors, faulty memory locations in hardware, or transmission over noisy digital links. For images, which are contaminated by impulse noise only, a nonsmooth  $L^1$  data-term is a more successful approach than a quadratic  $L^2$  term [1, 44, 45]. In this situation we have to solve an optimization problem of the form

$$\min_{u \in BV(\Omega)} \|Tu - g\|_{L^1(\Omega)} + \lambda \int_{\Omega} |Du|, \quad (1.2)$$

which we call  *$L^1$ -TV model*. Methods for obtaining a numerical solution of problem (1.2) might be found in [5, 11, 18].

Instead of assuming that an image is only corrupted by one type of noise, a more realistic approach considers a mixture of different types of noises, such as Gaussian noise and impulse noise. This might be the case if in the process of image acquisition, which describes the capturing of an image by a camera and converting it into a measurable entity [42], Gaussian noise occurs and later while image transmission impulse noise is added. In such a situation the choice of the data term is unclear. Most of the known approaches determinate or approximate outliers (impulse noise) in the image in the first phase and then in the second phase methods for Gaussian denoising are used [8, 28, 33, 39, 59, 62]. For example in [8], in the second phase a functional consisting of the Mumford–Shah regularizer [43], which renders the problem nonconvex, is minimized. In general Gaussian plus impulse noise removal strategies may be classified in the following way: filter approaches [26, 47, 62], regularization based approaches [8, 17, 28, 33, 39, 51, 59, 61], Bayesian-based approaches [41] and patch-based approaches [16, 38, 40].

In Chapter 2, containing the results published in [AL1], we propose a new model, or optimization problem, for simultaneously removing Gaussian and impulse noise. The objective of the model is composed by a total variation term (as a regularizer) and a combination of a quadratic  $L^2$  term and a nonsmooth  $L^1$  term, which renders the problem convex. That is we propose

$$\min_{u \in BV(\Omega)} \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \int_{\Omega} |Du|, \quad (1.3)$$

where  $T_i : L^2(\Omega) \rightarrow L^2(\Omega)$  is a bounded linear operator,  $g_i \in L^2(\Omega)$  is the given data,  $\alpha_i \geq 0$  with  $\alpha_1 + \alpha_2 > 0$  for  $i = 1, 2$ . We call this approach the  *$L^1$ - $L^2$ -TV model*. Numerical experiments demonstrate that such a combined data term is indeed well suited to the task of removing simultaneously Gaussian and impulse noise and yields visually more appealing

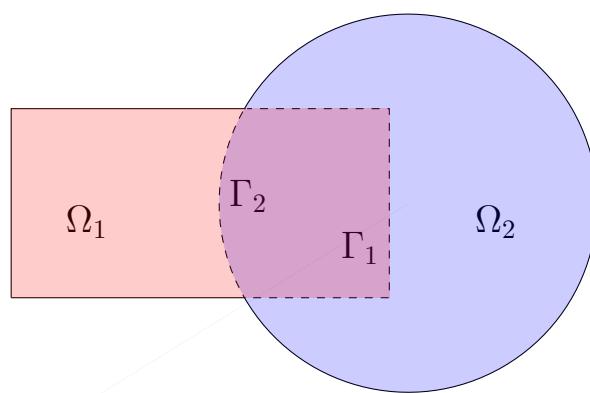
## I. Subspace Correction Methods for Total Variation Minimization

results than conventional methods. It is even shown analytically by a concrete example, see [AL1, Example 2.1], that the minimizer of the newly proposed functional has remarkable advantages over the  $L^2$ -TV and  $L^1$ -TV models.

All the aforementioned methods for total variation minimization are restricted to small- or medium-size problems. For large problems, as they occur in nearly all applications nowadays, subspace correction methods are required. Subspace correction methods are “divide and conquer” techniques practically most valuable to numerically solve partial differential equations [60]. The idea of these methods are to divide the global problem into several smaller problems and to iteratively solve the appropriately defined subproblems on each subspace.

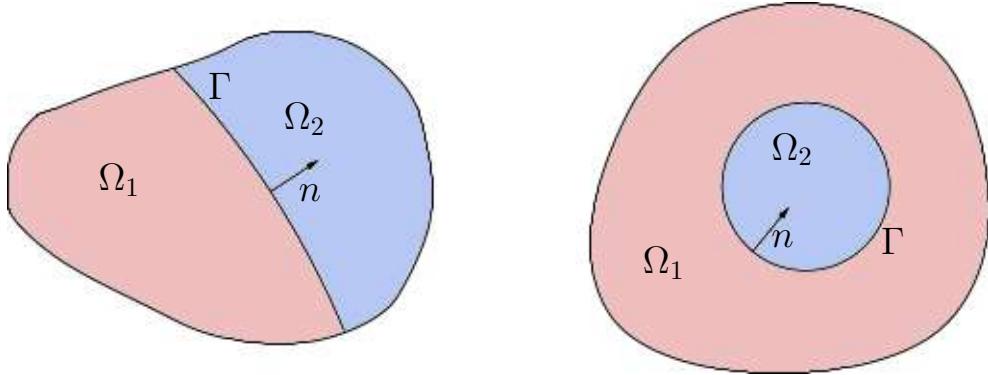
From a practical point of view one of the appeals of subspace correction methods is, that they allow to develop parallel algorithms, which are able to use the efficiency of multi-processors and cluster environments. Such parallel approaches are necessary, if one wants to solve high-dimensional numerical problems, as they appear in a wide range of applications in physics and engineering. The main advantages of such an approach are (i) reduction of dimension; (ii) extension of parallelisation; (iii) local treatment of complex and irregular geometries, singularities and abnormal domains; and sometimes (iv) reduction of the numerical complexity of the underlying solution strategy. One of the most important representatives of this class of algorithms are the Jacobi method, the Gauss–Seidel method, relaxation methods, multigrid methods and domain decomposition methods. These methods can often be directly applied to the partial differential equation of interest, but also the discretization of the problem is of major interest. For more details on subspace correction methods and the associated solvers we refer to [60].

H. A. Schwarz [50] proposed the first known subspace correction method. More precisely, he developed an overlapping domain decomposition method in order to show the existence of harmonic functions on irregular domains, which are the union of overlapping subdomains [48, p. 26]. Domain decomposition refers to the separation of a spatial domain into several subregions. The original problem is then tackled iteratively by alternately solving the constituted problems on the subdomains.



**Figure 1.2:** Overlapping decomposition of the physical domain into two domains.

When we talk about domain decomposition methods we distinguish between overlapping (Figure 1.2) and nonoverlapping (Figure 1.3) of the physical domain into two or more sub-



**Figure 1.3:** Nonoverlapping decomposition of the physical domain into two domains.

mains. Sophisticated techniques, such as “coloring”, are often used, when a partitioning into a larger amount of subdomains is performed [13, 48, 54]. This splitting strategy can be adapted to general subspace correction methods. There, instead of partitioning the spatial domain, the function space, on which the underlying problem is defined, is separated. In a variational setting this means, that on each subspace an energy is minimized. As long as the energy is smooth (differentiable), it is well understood how convergent subspace correction methods can be constructed, whose limit is a solution of the original global problem. In such a case usually the convergence rate and the independence of the convergence from the mesh size of discretization are well established.

For nonsmooth problems the resulting splitting algorithms still work fine, as long as the energy is additive with respect to the subspace decomposition. For such problems, the convergence to the solution of the global problem is ensured and sometimes even the convergence rate can be derived, see [21, 56]. For example, in [57] for the application of image deblurring even preconditioning effects of a specific subspace correction algorithm of minimizing a non-smooth, (but) additive energy are shown. However, for nonsmooth and nonadditive energies the research on subspace correction methods is far from being complete and only very few is known till now. For some problem classes even counterexamples exist, indicating failure of subspace correction, see [22, 58].

As we have seen above, in image reconstruction one might be particularly interested in minimizing nonsmooth and nonadditive energies, as problems (1.1) and (1.2). Note, that the total variation is nonsmooth and nonadditive. Due to the continuous improvement of hardware, the dimensionality of images and measured data in general is permanently increasing. This results in huge data sets, which need to be further processed, since noise and/or other type of corruptions are present in the data. In order to handle the resulting large problems in reasonable time, subspace correction and domain decomposition methods are needed, since they allow to separate the workload. Then instead of one large problem a sequence of smaller problems, which are easier to handle, are solved. Additionally, this partitioning allows to distribute the workload on multiple cores with the possibility of parallelization.

The major difficulty of total variation minimization in the context of domain decomposition lies in the correct treatment of the interfaces of the domain patches, i.e., the preservation of crossing discontinuities and the correct matching of the solution where it is continuous. We emphasize that well-known approaches such as those in [9, 13, 53, 55] are not (theoretically) guaranteed to work successfully for total variation minimization and thus are not directly

## I. Subspace Correction Methods for Total Variation Minimization

applicable to nonsmooth and nonadditive minimization problems.

In [23, 24, 25] first nonoverlapping and overlapping domain decomposition methods for solving the  $L^2$ -TV model are introduced. In particular, in [24, 25] a convex objective functional with linear constraints, which ensure the correct treatment of the interfaces, is iteratively minimized on each subdomain. While the implementations in these two papers guarantee the monotonic decay of the energy and the convergence of the method, its convergence to a solution of the global problem (1.1) is analytically in general not ensured. In [24] a proof of convergence of overlapping domain decomposition methods is presented, which shows in a discrete and one-dimensional setting that the proposed methods indeed approach the solution of the original global problem. However, it is not clear yet, how to extend the proof to a higher-dimensional setting without imposing quite restrictive assumptions.

In [22] a wavelet decomposition method is presented with properties similar to those of the aforementioned domain decomposition methods. Imposing an additional condition, allows to establish global optimality of the limit point obtained by the domain decomposition methods. Unfortunately this additional condition in general does not always hold. In [22] even a counterexample is constructed, showing that subspace correction methods in general do not converge to a solution of the global problem.

For the minimization of problem (1.1) domain decomposition methods are developed in [19] for image segmentation by graph cuts and in [36] based on a primal-dual stitching. Further in [15] splitting methods for the minimization of the nonlocal total variation are constructed. However, for the decomposition approaches in [15, 19, 36] no rigorous theoretical analysis is provided.

This part of the thesis is organized as follows. In Chapter 2, containing the results of [**AL1**], for the following more general class of nonsmooth and nonadditive convex variational problems

$$\min_{u \in BV(\Omega)} \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \int_{\Omega} \varphi(|Du|), \quad (1.4)$$

where  $\varphi(|\cdot|)$  is a convex function of measures representing the regularizer, sequential and parallel overlapping and nonoverlapping domain decomposition methods are presented. For  $\varphi(x) = x$  the regularizer becomes the total variation and model (1.4) the  $L^1$ - $L^2$ -TV model. Hence, we note that the  $L^2$ -TV model, considered in [23, 24, 25], and the  $L^1$ -TV model are special instances of model (1.4). The subspace correction methods presented in [**AL1**] are shown to be convergent and it is theoretically ensured that the global energy is monotonically decreasing within the iterations. Additionally, we derive in [**AL1**] an estimate of the distance between the true solution of the global problem (1.4) and the limit point obtained from the subspace correction method. With the help of this estimate in our numerical experiments we demonstrate that the proposed subspace correction methods indeed approach the minimizer of the global objective functional.

The subspace correction methods developed in [**AL1**] request to solve an optimization problem on each subdomain exactly. In practice this is not always possible. Therefore, we develop in Chapter 3, see also [**AL2**], subspace correction methods for model (1.4), whereas on each subspace an easier manageable auxiliary functional is optimized. In this case, as in [**AL1**], we are able to prove convergence and the monotonic decay of the global energy. Moreover, we also achieve an estimate for the distance of the computed solution to the real minimizer of the global problem. This estimate allows us again to show in our numerical experiments that the proposed methods indeed generate sequences which converge to the true minimizer of the global problem.

## 1. Introduction and Overview

In Chapter 4, containing the results of [AL3], we follow a different approach of constructing convergent domain decomposition methods for total variation based imaging problems. In particular, instead of constructing a domain decomposition method for the nonsmooth and nonadditive  $L^2$ -TV problem we consider a predual problem of (1.1). Note, that for  $T = I$  (identity) a predual problem of problem (1.1) may be written as [31]

$$\begin{aligned} \min \frac{1}{2} \|\operatorname{div} \mathbf{p} + g\|_{L^2(\Omega)}^2 & \quad \text{over} \quad \mathbf{p} \in H_0(\operatorname{div}, \Omega) \\ \text{subject to (s.t.) } -\lambda \mathbf{1} \leq \mathbf{p}(x) \leq \lambda \mathbf{1} & \quad \text{for almost all (f.a.a.) } x \in \Omega, \end{aligned} \tag{1.5}$$

where  $H_0(\operatorname{div}, \Omega) := \{\mathbf{v} \in (L^2(\Omega))^2 : \operatorname{div} \mathbf{v} \in L^2(\Omega), \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega\}$ ,  $\mathbf{n}$  is the outward unit normal on  $\partial\Omega$ , and  $\mathbf{1}$  denotes the constant vector with entry values 1. We remark, that the objective functional in (1.5) is neither strictly nor strongly convex. Thus known approaches, as those in [52, 53, 56], are not directly applicable to problem (1.5). Nevertheless, due to the smooth objective functional and the pointwise constraints, problem (1.5) seems to be more amenable to domain decomposition than the structure of (1.1).

In fact, for problem (1.5) we develop sequential and parallel nonoverlapping domain decomposition methods [AL3]. In a discrete setting we prove analytically the convergence of the proposed splitting strategies to a solution of the global problem (1.5). Further, we compare numerically this dual splitting strategy with the primal domain decomposition approach from [AL1]. We observe, that the dual approach converges much faster to a solution, requiring significantly less iterations, than the primal method. In this vein, the algorithms in [AL3] are not only analytically but also practically superior to the ones in [AL1]. Although the theoretical analysis in [AL3] is only performed for the case  $T = I$ , it can be easily extended to more general cases, for example, to the case where  $T$  is a convolution operator, see [AL3, Remark 2].

Similarly in [14] overlapping domain decomposition methods for the predual total variation formulation are developed. In particular, these methods are even shown to converge with a certain convergence rate to a solution of the global problem in an infinite dimensional setting.

Based on the nonoverlapping domain decomposition methods of [AL3] in [37] nonoverlapping domain decomposition algorithms are constructed for a discretized version of the primal problem (1.1), using the following strategy: The domain decomposition method in [AL3] is constituted by its subdomain problems. Then the dual problems of these subdomain problems are computed, yielding a sequence of nonoverlapping subdomain problems of the primal problem in a discrete setting. The connection between the primal and predual problems, allows in [37] to show analytically the convergence of the proposed nonoverlapping splitting methods to the minimizer of the global problem in a finite difference setting.

Following the idea of [37], we construct in Chapter 5, see [AL4], overlapping domain decomposition methods for the infinite dimensional total variation minimization problem (1.1) with  $T = I$ . That is, we consider the nonoverlapping domain decomposition method of [14] for the predual problem, which is given in a continuous setting, and compute from their subdomain problems the respective dual formulation. This allows us to construct an overlapping decomposition method for the primal total variation minimization problem. The (strong) convergence of these methods to a minimizer of the global problem is guaranteed in an infinite dimensional setting. Note, that the analysis presented in [AL4] differs significantly from the one in [37]. Firstly, in contrast to [37] we consider overlapping domain decomposition methods, and hence our subdomain problems are very different to the ones in [37]. Consequently

## I. Subspace Correction Methods for Total Variation Minimization

a different convergence analysis as well as different subdomain solvers have to be developed. Secondly, our analysis is carried out in an infinite dimensional setting, which generates additional difficulties in proving convergence. Note, that the domain decomposition methods presented in [AL4] are the first splitting methods for the minimization of the total variation with a theoretically guaranteed convergence to the global minimum in a continuous setting.

## References

- [1] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE Transactions on Signal Processing*, 45(4):913–917, 1997.
- [2] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, 2000.
- [3] G. Aubert and J.-F. Aujol. A variational approach to removing multiplicative noise. *SIAM Journal on Applied Mathematics*, 68(4):925–946, 2008.
- [4] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing*, volume 147 of *Applied Mathematical Sciences*. Springer, New York, second edition, 2006.
- [5] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition – modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67(1):111–136, 2006.
- [6] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.
- [7] M. Burger, A. Sawatzky, and G. Steidl. First order algorithms in variational image processing. In *Splitting Methods in Communication, Imaging, Science, and Engineering*, pages 345–407. Springer, 2016.
- [8] J.-F. Cai, R. H. Chan, and M. Nikolova. Two-phase approach for deblurring images corrupted by impulse plus Gaussian noise. *Inverse Problems and Imaging*, 2(2):187–204, 2008.
- [9] C. Carstensen. Domain decomposition for a non-smooth convex minimization problem and its application to plasticity. *Numerical Linear Algebra with Applications*, 4(3):177–190, 1997.
- [10] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock. An introduction to total variation for image analysis. *Theoretical Foundations and Numerical Methods for Sparse Recovery*, 9:263–340, 2010.
- [11] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [12] R. H. Chan, C.-W. Ho, and M. Nikolova. Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. *IEEE Transactions on Image Processing*, 14(10):1479–1485, 2005.
- [13] T. F. Chan and T. P. Mathew. Domain decomposition algorithms. *Acta Numerica*, 3:61–143, 1994.
- [14] H. Chang, X.-C. Tai, L.-L. Wang, and D. Yang. Convergence rate of overlapping domain decomposition methods for the Rudin–Osher–Fatemi model based on a dual formulation. *SIAM Journal on Imaging Sciences*, 8(1):564–591, 2015.
- [15] H. Chang, X. Zhang, X.-C. Tai, and D. Yang. Domain decomposition methods for nonlocal total variation image restoration. *Journal of Scientific Computing*, 60(1):79–100, 2014.

## 1. Introduction and Overview

- [16] J. Delon and A. Desolneux. A patch-based approach for removing impulse or mixed Gaussian-impulse noise. *SIAM Journal on Imaging Sciences*, 6(2):1140–1174, 2013.
- [17] B. Dong, H. Ji, J. Li, Z. Shen, and Y. Xu. Wavelet frame based blind image inpainting. *Applied and Computational Harmonic Analysis*, 32(2):268–279, 2012.
- [18] Y. Dong, M. Hintermüller, and M. Neri. An efficient primal-dual method for  $L^1$  TV image restoration. *SIAM Journal on Imaging Sciences*, 2(4):1168–1189, 2009.
- [19] Y. Duan and X.-C. Tai. Domain decomposition methods with graph cuts algorithms for total variation minimization. *Advances in Computational Mathematics*, 36(2):175–199, 2012.
- [20] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*, volume 375 of *Mathematics and its Applications*. Kluwer Academic Publishers Group, Dordrecht, 1996.
- [21] M. Fornasier. Domain decomposition methods for linear inverse problems with sparsity constraints. *Inverse Problems*, 23(6):2505, 2007.
- [22] M. Fornasier, Y. Kim, A. Langer, and C.-B. Schönlieb. Wavelet decomposition method for  $L_2$ /TV-image deblurring. *SIAM Journal on Imaging Sciences*, 5(3):857–885, 2012.
- [23] M. Fornasier, A. Langer, and C.-B. Schönlieb. Domain decomposition methods for compressed sensing. In *Proceedings of the International Conference of SampTA09, Marseilles*, 2009.
- [24] M. Fornasier, A. Langer, and C.-B. Schönlieb. A convergent overlapping domain decomposition method for total variation minimization. *Numerische Mathematik*, 116(4):645–685, 2010.
- [25] M. Fornasier and C.-B. Schönlieb. Subspace correction methods for total variation and  $l_1$ -minimization. *SIAM Journal on Numerical Analysis*, 47(5):3397–3428, 2009.
- [26] R. Garnett, T. Huegerich, C. Chui, and W. He. A universal noise removal algorithm with an impulse detector. *IEEE Transactions on Image Processing*, 14(11):1747–1754, 2005.
- [27] P. Getreuer, M. Tong, and L. A. Vese. A variational model for the restoration of MR images corrupted by blur and Rician noise. In *Advances in Visual Computing*, pages 686–698. Springer, 2011.
- [28] O. Ghita and P. F. Whelan. A new GVF-based image enhancement formulation for use in the presence of mixed noise. *Pattern Recognition*, 43(8):2646–2658, 2010.
- [29] G. Gilboa and S. Osher. Nonlocal operators with applications to image processing. *Multiscale Modeling & Simulation*, 7(3):1005–1028, 2008.
- [30] E. Giusti. *Minimal Surfaces and Functions of Bounded Variation*, volume 80 of *Monographs in Mathematics*. Birkhäuser Verlag, Basel, 1984.
- [31] M. Hintermüller and K. Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM Journal on Applied Mathematics*, 64(4):1311–1333, 2004.
- [32] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.
- [33] Y.-M. Huang, M. K. Ng, and Y.-W. Wen. Fast image restoration methods for impulse and Gaussian noises removal. *IEEE Signal Processing Letters*, 16(6):457–460, 2009.
- [34] S. Kindermann, S. Osher, and P. W. Jones. Deblurring and denoising of images by nonlocal functionals. *Multiscale Modeling & Simulation*, 4(4):1091–1115, 2005.
- [35] T. Le, R. Chartrand, and T. J. Asaki. A variational approach to reconstructing images corrupted by Poisson noise. *Journal of Mathematical Imaging and Vision*, 27(3):257–263, 2007.
- [36] C.-O. Lee, J. H. Lee, H. Woo, and S. Yun. Block decomposition methods for total variation by primal-dual stitching. *Journal of Scientific Computing*, 68(1):273–302, Jul 2016.

## I. Subspace Correction Methods for Total Variation Minimization

- [37] C.-O. Lee and C. Nam. Primal domain decomposition methods for the total variation minimization, based on dual decomposition. *SIAM Journal on Scientific Computing*, 39(2):B403–B423, 2017.
- [38] B. Li, Q. Liu, J. Xu, and X. Luo. A new method for removing mixed noises. *Science China Information Sciences*, 54(1):51–59, 2011.
- [39] Y.-R. Li, L. Shen, D.-Q. Dai, and B. W. Suter. Framelet algorithms for de-blurring images corrupted by impulse plus Gaussian noise. *IEEE Transactions on Image Processing*, 20(7):1822–1837, 2011.
- [40] J. Liu, X.-C. Tai, H. Huang, and Z. Huan. A weighted dictionary learning model for denoising images corrupted by mixed noise. *IEEE Transactions on Image Processing*, 22(3):1108–1120, 2013.
- [41] E. López-Rubio. Restoration of images corrupted by Gaussian and uniform impulsive noise. *Pattern Recognition*, 43(5):1835–1846, 2010.
- [42] T. B. Moeslund. *Introduction to Video and Image Processing: Building Real Systems and Applications*. Springer Science & Business Media, 2012.
- [43] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.
- [44] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40(3):965–994, 2002.
- [45] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1-2):99–120, 2004.
- [46] K. Papafitsoros and C.-B. Schönlieb. A combined first and second order variational approach for image reconstruction. *Journal of Mathematical Imaging and Vision*, 48(2):308–338, 2014.
- [47] S. Peng and L. Lucke. Fuzzy filtering for mixed noise removal during image processing. In *Proceedings of the Third IEEE Conference on Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence*, pages 89–93. 1994.
- [48] A. Quarteroni and A. Valli. *Domain decomposition methods for partial differential equations*. Oxford University Press, 1999.
- [49] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.
- [50] H. A. Schwarz. Über einige Abbildungsaufgaben. *Journal für die Reine und Angewandte Mathematik*, 70:105–120, 1869.
- [51] Y. Shen, B. Han, and E. Braverman. Removal of mixed Gaussian and impulse noise using directional tensor product complex tight framelets. *Journal of Mathematical Imaging and Vision*, pages 1–14, 2015.
- [52] X.-C. Tai. Rate of convergence for some constraint decomposition methods for nonlinear variational inequalities. *Numerische Mathematik*, 93(4):755–786, 2003.
- [53] X.-C. Tai and P. Tseng. Convergence rate analysis of an asynchronous space decomposition method for convex minimization. *Mathematics of Computation*, 71(239):1105–1135, 2002.
- [54] A. Toselli and O. B. Widlund. *Domain decomposition methods: algorithms and theory*, volume 34. Springer, 2005.
- [55] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications*, 109(3):475–494, 2001.

## 1. Introduction and Overview

- [56] P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117(1):387–423, 2009.
- [57] C. Vonesch and M. Unser. A fast multilevel algorithm for wavelet-regularized image restoration. *IEEE Transactions on Image Processing*, 18(3):509–523, 2009.
- [58] J. Warga. Minimizing certain convex functions. *Journal of the Society for Industrial and Applied Mathematics*, 11:588–593, 1963.
- [59] Y. Xiao, T. Zeng, J. Yu, and M. K. Ng. Restoration of images corrupted by mixed Gaussian-impulse noise via  $l_1 - l_0$  minimization. *Pattern Recognition*, 44(8):1708–1720, 2011.
- [60] J. Xu. The method of subspace corrections. *Journal of Computational and Applied Mathematics*, 128(1):335–362, 2001.
- [61] M. Yan. Restoration of images corrupted by impulse noise and mixed Gaussian impulse noise using blind inpainting. *SIAM Journal on Imaging Sciences*, 6(3):1227–1245, 2013.
- [62] J. X. Yang and H. R. Wu. Mixed Gaussian and uniform impulse noise analysis using robust estimation for digital images. In *16th International Conference on Digital Signal Processing, 2009*, pages 1–5, 2009.
- [63] X. Zhang, M. Burger, X. Bresson, and S. Osher. Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM Journal on Imaging Sciences*, 3(3):253–276, 2010.



## 2. Subspace Correction Methods for a Class of Nonsmooth and Nonadditive Convex Variational Problems

**Bibliographic Note:** The content of this chapter is published in the article [AL1]: M. Hintermüller and A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1/L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.

**Summary:** The minimization of a functional composed of a nonsmooth and nonadditive regularization term and a combined  $L^1$  and  $L^2$  data-fidelity term is proposed. It is shown analytically and numerically that the new model has noticeable advantages over popular models in image processing tasks. For the numerical minimization of the new objective, subspace correction methods are introduced which guarantee the convergence and monotone decay of the associated energy along the iterates. Moreover, an estimate of the distance between the outcome of the subspace correction method and the global minimizer of the nonsmooth objective is derived. This estimate and numerical experiments for image denoising, inpainting, and deblurring indicate that in practice the proposed subspace correction methods indeed approach the global solution of the underlying minimization problem.

### 2.1. Introduction

Subspace correction is a divide-and-conquer technique originally proposed for the numerical solution of partial differential equations. Algorithmically this is achieved by iteratively solving on each subspace an appropriately defined subproblem, which, in a variational setting, typically amounts to minimizing a smooth energy. For the overall algorithm, convergence, rate of convergence, and the independence of the rate of convergence from the mesh size of discretization are well established.

For nonsmooth problems, the resulting splitting algorithms still work fine as long as the energy splits additively with respect to the subspace decomposition. For such problems convergence and sometimes even rate of convergence are ensured; see e.g. [27, 49]. Moreover, for image deblurring problems preconditioning effects of a specific subspace correction algorithm for minimizing a nonsmooth energy are shown in [51]. For nonsmooth and *nonadditive* energies, however, the research on subspace correction methods is far from being complete, and for some problem classes counterexamples exist indicating failure of subspace correction; see e.g. [28, 52].

From a computational point of view, one of the appeals of subspace correction methods is the fact that parallel algorithms can be devised which exploit the capabilities of multiprocessor or multicore computer architectures. Main advantages of associated iterative solvers include (i) dimension reduction; (ii) enhancement of parallelism; (iii) localized treatment of complex

## I. Subspace Correction Methods for Total Variation Minimization

and irregular geometries, singularities, and anomalous regions; and sometimes (iv) reduction of the computational complexity of the underlying solution method. Among the important representatives of this algorithm class one finds the Jacobi method, the Gauss–Seidel method, point or block relaxation methods, multigrid methods, and domain decomposition methods. For further details on subspace correction and associated solvers, we refer the reader to [54].

In this paper we focus on subspace correction methods for a class of nonsmooth and nonadditive problems which arise in mathematical image processing. In this area the importance of devising such methods is clearly motivated by the continuous improvement of imaging hardware, which allows one to increase resolutions or to acquire vast amounts of data. In the context of variational methods in image processing, this may lead to extremely large-scale problems which need to be processed routinely.

In image restoration, the nonsmooth and nonadditive total variation (TV), proposed in [45] for image denoising, plays a fundamental role as a regularization technique, since it preserves edges and discontinuities in images. In this context, one typically minimizes an energy that consists of a data-fidelity term, which enforces the consistency between the recovered and the measured image, and the total variation as the regularization term. The choice of the data term usually depends on the type of noise contained in the measured image data. In this vein, for images corrupted by *Gaussian noise* a quadratic  $L^2$  data-fidelity term has been successfully used in first order methods (see e.g. [11, 12, 13, 17, 19, 20, 21, 22, 24, 32, 40, 43, 53, 57]) as well as in second order methods (see e.g. [34]). In this approach, which we refer to as the  $L^2$ -TV model, the image  $u$  is recovered from the observed data  $g$  by solving

$$\min_u \alpha \|Tu - g\|_{L^2(\Omega)}^2 + |Du|(\Omega), \quad (2.1)$$

where  $\Omega \subset \mathbb{R}^2$  is an open bounded set with Lipschitz boundary,  $T$  is a bounded linear operator modeling the image-formation device (if the image is only corrupted by noise, one sets  $T = I$ ), and  $\alpha > 0$  is a parameter. We recall that for  $u \in L^1(\Omega)$

$$V(u, \Omega) := \sup \left\{ \int_{\Omega} u \operatorname{div} \phi dx : \phi \in [C_c^1(\Omega)]^2, \|\phi\|_{\infty} \leq 1 \right\}$$

is the variation of  $u$ . In the event that  $V(u, \Omega) < \infty$  we denote  $|Du|(\Omega) = V(u, \Omega)$  and call it the total variation of  $u$  in  $\Omega$ ; see [2] for more details. If  $u \in W^{1,1}(\Omega)$ , then  $|Du|(\Omega) = \int_{\Omega} |\nabla u| dx$ . The  $L^2$ -TV model usually does not yield a satisfactory restoration in the presence of *salt-and-pepper noise*, where the noisy image  $g$ , assumed throughout to have a dynamic range of  $c_{\min} \leq g \leq c_{\max}$ , is given by

$$g(x) = \begin{cases} c_{\min} & \text{with probability } p_1 \in [0, 1), \\ c_{\max} & \text{with probability } p_2 \in [0, 1), \\ u(x) & \text{with probability } 1 - p_1 - p_2, \end{cases}$$

with  $1 - p_1 - p_2 > 0$  [15]. Here,  $p_1 + p_2$  defines the noise level. Recently a nonsmooth  $L^1$  data-fidelity term was suggested in [1], which treats impulse noise (e.g. salt-and-pepper noise) more successfully than a quadratic  $L^2$  data term [41, 42, 25]; i.e., instead of (2.1), one considers

$$\min_u \alpha \|Tu - g\|_{L^1(\Omega)} + |Du|(\Omega),$$

which we call the  $L^1$ -TV model.

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

In the case of simultaneous Gaussian and salt-and-pepper noise the choice of the data-fidelity is unclear, and the literature on this subject appears rather scarce. In order to accommodate such situations, a two-phase reconstruction approach is suggested in [9]. In fact, in the first phase (most of) the outliers are detected, and in the second phase a variational functional consisting of a Mumford–Shah regularizer, which renders the problem nonconvex, is minimized. In contrast to this development we tackle the problem of removing simultaneous Gaussian and salt-and-pepper noise by optimizing a convex functional with a total variation regularizer and a combination of a quadratic  $L^2$ -term and a nonsmooth  $L^1$ -term. It turns out in our numerical experiments that such a combined data-fidelity term well suits the restoration task; see Figure 2.2 below. Analytically we show by means of an explicit example that the minimization of the newly proposed functional has noticeable advantages over the standard functionals, i.e., the  $L^2$ -TV and  $L^1$ -TV model. Algorithmically, we adapt the approach in [4], which was originally proposed for solving the  $L^1$ -TV model only, to our case of a combined data-fidelity term.

As all of the aforementioned solvers for TV-minimization are confined to small- and medium-scale problems only, we propose and analyze subspace correction, domain decomposition, and coordinate descent methods, as these are fundamental for reducing the overall problem to a finite number of subproblems, each of a size manageable for the above TV solvers.

Recently, in [28, 29, 30], nonoverlapping and overlapping domain decomposition strategies were introduced for solving the  $L^2$ -TV problem. In this context, the major difficulty lies in the correct treatment of the interfaces of the domain decomposition patches, i.e. the preservation of crossing discontinuities and the correct matching where the solution is continuous. We emphasize that well-known approaches such as those in [10, 18, 46, 47] are not directly applicable to the nonsmooth and nonadditive  $L^2$ -TV problem. In [29, 30] the convex objective under some linear constraint, ensuring the correct treatment of the internal interfaces, was iteratively minimized on each subdomain. While in these two papers an implementation guaranteeing convergence and monotonic decay of the objective energy is provided, convergence to the global minimizer of the  $L^2$ -TV problem *cannot* be ensured, in general. For one-dimensional problems, in [29] a proof is presented which establishes convergence of the overlapping domain decomposition algorithm to the global solution. We note that, although this proof is carried out for any finite-dimensional space, it is not yet clear how to prove convergence to the expected minimizer without further (and practically possibly critical) assumptions on the overlapping region for higher dimensions ( $d > 1$ ).

In [28] a wavelet decomposition method is presented with properties similar to those of the aforementioned nonoverlapping domain decomposition methods. In that paper, an additional condition is imposed which allows one to establish global optimality of the limit point obtained by the domain decomposition method. Unfortunately, despite the good practical behavior of the method, this condition cannot be ensured to hold in general, as counterexamples have shown. Thus, with the aforementioned condition one can only check *a posteriori* whether the algorithm found the global minimizer or failed to do so. Moreover, no error estimates are available.

In the present paper we generalize the subspace correction strategy to more general functionals, which consist of a nonsmooth and nonadditive regularization term and a weighted combination of an  $L^1$ -term and a quadratic  $L^2$ -term; see (2.2) below. In this setting, the  $L^2$ -TV model considered in [28, 29, 30] and the  $L^1$ -TV model are special instances. Note that the methods in [28, 29, 30] differ from our approach. In fact, in [28, 29, 30] each subspace minimization problem is approximated by a surrogate functional minimization, while we are minimizing on each subspace the exact subspace minimization problem. Thus, a different

## I. Subspace Correction Methods for Total Variation Minimization

convergence analysis is required. Similarly to the domain decomposition methods in [29, 30], we are able to show that our subspace correction methods for the newly introduced functional are guaranteed to converge and to monotonically decrease the energy. In addition, we are able to establish an estimate of the distance of the limit point obtained from the subspace correction method to the true global minimizer. With the help of this estimate, we demonstrate in our numerical experiments that the sequence generated by our proposed algorithm indeed approaches the global minimizer of the objective functional.

The rest of the paper is organized as follows: In Section 2.2 we state the problem of interest, and we propose a solution method for the global minimization problem. Moreover, we motivate the choice of the objective functional by analyzing theoretically an illustrative example as well by numerical experiments. Our alternating and parallel subspace correction methods are introduced in Section 2.3 in a Banach space setting where we also state some convergence properties. In Section 2.4 we describe the problem in a discrete setting and show optimality properties of our subspace correction methods which allow us to estimate the distance of a limit point obtained by subspace correction to the minimizer of the total energy. In Section 2.5 we present our subspace correction methods for the special cases of overlapping and nonoverlapping domain decomposition. Details on the implementation of the solvers for the domain decomposition methods are provided. Finally we show sequential and parallel numerical experiments for TV minimization.

## 2.2. Image Restoration with Mixed $L^1/L^2$ Data-Fidelity

We are interested in solving the following minimization problem:

$$\min_{u \in L^2(\Omega)} J_{\alpha_1, \alpha_2}(u) := \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \varphi(|Du|)(\Omega), \quad (2.2)$$

where  $T_i : L^2(\Omega) \rightarrow L^2(\Omega)$  is a bounded linear operator,  $g_i \in L^2(\Omega)$  is a given datum,  $\alpha_i \geq 0$  for  $i = 1, 2$ , with  $\alpha_1 + \alpha_2 > 0$ ,  $\Omega \subset \mathbb{R}^d$ ,  $d \in \mathbb{N}$ , and  $\varphi(|\cdot|)$  is a convex function of measures representing regularization.

We assume that  $\|T_i\| < 1$  for  $i = 1, 2$ , which is not at all a restriction, as a proper rescaling of the problem reestablishes the desired setting whenever a norm exceeds 1. In what follows we make the assumption that  $J_{\alpha_1, \alpha_2}$  is bounded from below and coercive, i.e.,  $\{J_{\alpha_1, \alpha_2} \leq C\} = \{u \in L^2(\Omega) : J_{\alpha_1, \alpha_2}(u) \leq C\}$  is bounded in  $L^2(\Omega)$  for all constants  $C > 0$ , in order to guarantee that problem (2.2) has solutions. Moreover we assume that

( $A_\varphi$ )  $\varphi : \mathbb{R} \rightarrow \mathbb{R}$  is a convex function, nondecreasing in  $\mathbb{R}^+$  with

- (i)  $\varphi(0) = 0$ ;
- (ii)  $cz - b \leq \varphi(z) \leq cz + b$  for all  $z \in \mathbb{R}^+$  for some constant  $c > 0$  and  $b \geq 0$ .

Note that for the particular example  $\varphi(t) = t$ , the third term in (2.2) becomes the well-known total variation of  $u$  in  $\Omega$ , and we then call (2.2) the  $L^1-L^2$ -TV model. Other functions which fulfill assumption ( $A_\varphi$ ) are  $\varphi(t) = \sqrt{1+t^2} - 1$  (the function of minimal surfaces) and  $\varphi(t) = \log \cosh t$  [50].

### 2.2.1. Qualitative Behavior of the $L^1$ - $L^2$ -TV Model

In order to motivate our proposed model (2.2), we use a simple and illustrative example in two dimensions, where  $\varphi(t) = t$ , which we compare with the  $L^1$ -TV model (i.e., when  $\alpha_2 = 0$

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

in (2.2)) and with the  $L^2$ -TV model (i.e., when  $\alpha_1 = 0$  in (2.2)).

**Example 2.1.** Let the observed image  $g_1 = g_2$  be the characteristic function  $1_{B_r(0)}$  of a disk  $B_r(0)$  centered at the origin with radius  $r > 0$ . We are interested in the explicit solution of the problem in (2.2) when  $\Omega = \mathbb{R}^2$ ,  $T_1 = T_2 = I$  is the identity operator, and  $\varphi(t) = t$  for three different cases: (i)  $\alpha_1 = 0$ ,  $\alpha_2 > 0$  ( $L^2$ -TV), (ii)  $\alpha_1 > 0$ ,  $\alpha_2 = 0$  ( $L^1$ -TV), (iii)  $\alpha_1 > 0$ ,  $\alpha_2 > 0$  ( $L^1$ - $L^2$ -TV).

For the first two cases we recall the solutions found in [16, 39].

(i) For  $\alpha_1 = 0$ ,  $\alpha_2 > 0$  the unique minimizer  $u_{0,\alpha_2}$  is given by

$$u_{0,\alpha_2} = \begin{cases} 0 & \text{if } 0 \leq r < \frac{1}{\alpha_2}, \\ \left(1 - \frac{1}{\alpha_2 r}\right) 1_{B_r(0)} & \text{if } r \geq \frac{1}{\alpha_2}. \end{cases}$$

(ii) For  $\alpha_1 > 0$ ,  $\alpha_2 = 0$  a minimizer  $u_{\alpha_1,0}$  is given by

$$u_{\alpha_1,0} \in \begin{cases} \{0\} & \text{if } 0 \leq r < \frac{2}{\alpha_1}, \\ \{c 1_{B_r(0)} : c \in [0, 1]\} & \text{if } r = \frac{2}{\alpha_1}, \\ \{1_{B_r(0)}\} & \text{if } r > \frac{2}{\alpha_1}. \end{cases}$$

(iii) For  $\alpha_1, \alpha_2 > 0$  one can reason that every minimizer has to be of the form  $c 1_{B_r(0)}$  for  $c \in [0, 1]$ . Therefore we just need to minimize the function

$$J_{\alpha_1, \alpha_2}(c 1_{B_r(0)}) = \alpha_1 \pi r^2 |1 - c| + \alpha_2 \pi r^2 (1 - c)^2 + 2\pi r c$$

over  $c \in [0, 1]$ . Then the optimality condition for  $c$  is given by

$$-\alpha_1 \pi r^2 - 2\alpha_2 \pi r^2 (1 - c) + 2\pi r = 0,$$

which is equivalent to

$$c = \frac{2\alpha_2 + \alpha_1}{2\alpha_2} - \frac{1}{\alpha_2 r}.$$

Hence, the unique minimizer is given by

$$u_{\alpha_1, \alpha_2} = \begin{cases} 0 & \text{if } 0 \leq r < \frac{2}{2\alpha_2 + \alpha_1}, \\ \left(\frac{2\alpha_2 + \alpha_1}{2\alpha_2} - \frac{1}{\alpha_2 r}\right) 1_{B_r(0)} & \text{if } \frac{2}{2\alpha_2 + \alpha_1} \leq r \leq \frac{2}{\alpha_1}, \\ 1_{B_r(0)} & \text{if } r > \frac{2}{\alpha_1}. \end{cases}$$

From this example we clearly see the difference between the  $L^2$ -TV model and the  $L^1$ -TV model. When the  $L^1$ -fidelity is used, then the solution is constant except at a special value ( $r = \frac{2}{\alpha_1}$ ), where it undergoes a sudden transition. When in addition to the  $L^1$ -fidelity also the  $L^2$ -term is present, then the solution is constant except in an interval ( $\frac{2}{2\alpha_2 + \alpha_1} \leq r \leq \frac{2}{\alpha_1}$ ) where it experiences a smooth transition. In contrast, when only the  $L^2$ -fidelity plus a TV-term is used, then the solution is only constant for  $0 \leq r < \frac{2}{\alpha_2}$  and hyperbolically increasing otherwise.

The differences between the  $L^2$ -TV model and the  $L^1$ -TV model result in the following observation: Fix  $\alpha_1 = \alpha_2 = \alpha > 0$  and set, as in Example 2.1,  $g_1 = g_2 = 1_{B_r(0)}$  and

## I. Subspace Correction Methods for Total Variation Minimization

$T_1 = T_2 = I$ . Then the solution  $u_{0,\alpha}$  of the  $L^2$ -TV model is identically 0 if  $r < \frac{1}{\alpha}$ . This is clearly an advantage over the  $L^1$ -TV model, where  $u_{\alpha,0} = 0$  if  $r < \frac{2}{\alpha}$ , since smaller features can be maintained with the  $L^2$ -TV model. In contrast, the  $L^2$ -TV model is not able to preserve the original features perfectly (except if  $\alpha = \infty$ ) but obtains them only with a loss of energy; i.e.,  $u_{0,\alpha} = (1 - \frac{1}{\alpha r}) 1_{B_r(0)}$  if  $r \geq \frac{1}{\alpha}$ . This is different for the  $L^1$ -TV model, where we have that  $u_{\alpha,0} = 1_{B_r(0)}$  if  $r > \frac{2}{\alpha}$ , and hence features can be perfectly preserved. This is naturally a clear advantage of the latter model.

For the combined  $L^1$ - $L^2$ -TV model, we observe that  $u_{\alpha,\alpha} = 0$  if  $0 \leq r < \frac{2}{3\alpha}$ , and hence even smaller features as with the  $L^2$ -TV model can be maintained. But still we are able to preserve original features perfectly as in the  $L^1$ -TV model. Moreover, the transition between the just mentioned constant states is smooth, which is clearly a property coming from the  $L^2$ -term, which renders the solution unique.

### 2.2.2. Practical Behavior

We first specify a solution algorithm for the model in (2.2) and then study the quantitative behavior of this model and the proposed method by means of a benchmark example.

**A Solution Algorithm** For computing a minimizer of the global problem in (2.2) we suggest an algorithm which is an adaptation of a method that was originally proposed for  $L^1$ -TV minimization problems in [4]. For this purpose we replace  $J_{\alpha_1, \alpha_2}$  by the functional

$$\alpha_1 \|v\|_{L^1(\Omega)} + \frac{1}{2\gamma} \|T_1 u - g_1 - v\|_{L^2(\Omega)}^2 + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \varphi(|Du|)(\Omega), \quad (2.3)$$

where  $\gamma > 0$  is small, so that we have  $g_1 \approx T_1 u - v$ . Actually, for  $\gamma \rightarrow 0$ , (2.3) approaches the objective functional in (2.2). Now we minimize (2.3) with respect to  $u$  and  $v$ , which we perform in the following alternating way:

(1) For fixed  $u$  solve

$$\min_{v \in L^2(\Omega)} \alpha_1 \|v\|_{L^1(\Omega)} + \frac{1}{2\gamma} \|T_1 u - g_1 - v\|_{L^2(\Omega)}^2. \quad (2.4)$$

The minimizer  $v^*$  of (2.4) can be easily computed via a soft-thresholding, i.e.,  $v^* = \text{ST}(T_1 u - g_1, \gamma \alpha_1)$ , where

$$\text{ST}(g, \beta)(x) = \begin{cases} g(x) - \beta & \text{if } g(x) > \beta, \\ 0 & \text{if } |g(x)| \leq \beta, \\ g(x) + \beta & \text{if } g(x) < -\beta \end{cases} \quad (2.5)$$

for all  $x \in \Omega$ .

(2) For fixed  $v$  solve

$$\min_{u \in L^2(\Omega)} \frac{1}{2\gamma} \|T_1 u - g_1 - v\|_{L^2(\Omega)}^2 + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \varphi(|Du|)(\Omega). \quad (2.6)$$

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

This step is realized on the *surrogate functional*

$$\begin{aligned} S(u, a) &:= \frac{1}{2\gamma} \|T_1 u - g_1 - v\|_{L^2(\Omega)}^2 + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \varphi(|Du|)(\Omega) \\ &\quad + \frac{1}{2\gamma} \left( \|u - a\|_{L^2(\Omega)}^2 - \|T_1(u - a)\|_{L^2(\Omega)}^2 \right) \\ &\quad + \alpha_2 \left( \|u - a\|_{L^2(\Omega)}^2 - \|T_2(u - a)\|_{L^2(\Omega)}^2 \right) \\ &= \frac{1}{2\gamma} \|u - z_1\|_{L^2(\Omega)}^2 + \alpha_2 \|u - z_2\|_{L^2(\Omega)}^2 + \varphi(|Du|)(\Omega) + \psi, \end{aligned}$$

with  $a, u \in L^2(\Omega)$  and where  $z_1 = z_1(a) = a + T_1^*(g_1 + v - T_1 a)$ ,  $z_2 = z_2(a) = a + T_2^*(g_2 - T_2 a)$ , and  $\psi$  is a function independent of  $u$ . Note that

$$\min_{u \in L^2(\Omega)} S(u, a) \Leftrightarrow \min_{u \in L^2(\Omega)} \left\| u - \frac{\gamma}{1 + 2\alpha_2\gamma} \left( \frac{1}{\gamma} z_1 + 2\alpha_2 z_2 \right) \right\|_{L^2(\Omega)}^2 + \frac{2\gamma}{1 + 2\alpha_2\gamma} \varphi(|Du|)(\Omega). \quad (2.7)$$

For  $\varphi(t) = t$ , (2.7) is a variant of the Rudin-Osher-Fatemi (ROF)-problem [45]. There exist several numerical methods for solving the ROF-problem efficiently; see, for example, [11, 17, 19, 22, 24, 32, 33, 34, 40, 43]. Hence an approximate solution of (2.6) can be computed by the following iterative algorithm: Initialize  $u^{(0)} \in L^2(\Omega)$  and iterate

$$u^{(\ell+1)} = \arg \min_{u \in L^2(\Omega)} S(u, u^{(\ell)}), \quad \ell \geq 0. \quad (2.8)$$

If  $T_1 = T_2 = I$  and  $\varphi(t) = t$ , then (2.6) becomes the ROF-problem, i.e.,

$$\min_{u \in L^2(\Omega)} \left\| u - \frac{\gamma}{1 + 2\alpha_2\gamma} \left( \frac{1}{\gamma} (g_1 + v) + 2\alpha_2 g_2 \right) \right\|_{L^2(\Omega)}^2 + \frac{2\gamma}{1 + 2\alpha_2\gamma} |Du|(\Omega),$$

which can be solved directly by means of the aforementioned methods.

**Numerical Examples** In Example 2.1 above we compute the exact solution of the minimization problem (2.2) with  $\varphi(t) = t$ . There we show that the newly proposed  $L^1$ - $L^2$ -TV model better preserves the original signal than do either the  $L^1$ -TV or the  $L^2$ -TV models. In this section we support this result by numerical computations for different choices of  $\alpha_1$  and  $\alpha_2$  in (2.2) for  $\varphi(t) = t$  and for a given noisy image  $g$  ( $= g_1 = g_2$ ), which is specified below. Note that the dynamic range of all image data considered in this paper is  $[c_{\min}, c_{\max}] := [0, 1]$ . As a comparison for the different restoration qualities of the image we use the PSNR (peak signal-to-noise ratio) given by

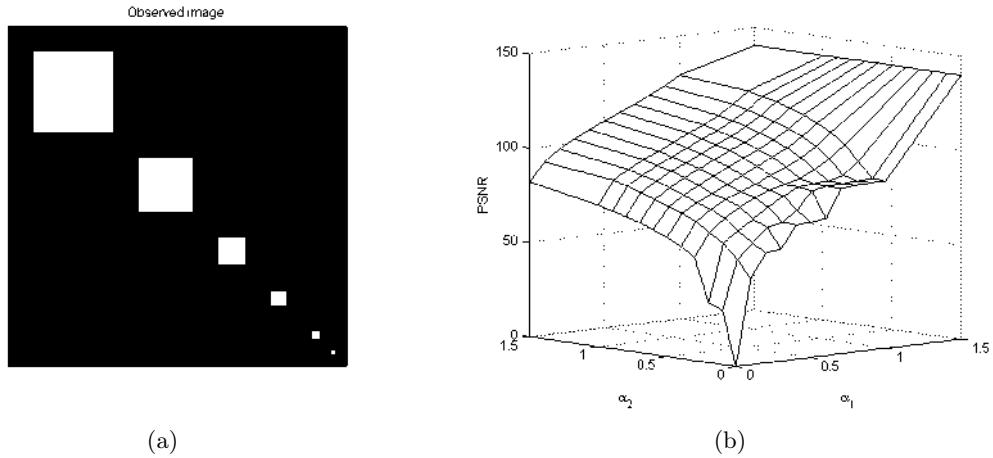
$$\text{PSNR} = 20 \log \frac{1}{\|u_{org} - u^*\|},$$

where  $u_{org}$  denotes the original image before any corruption and  $u^*$  the restored image. In general, when comparing PSNR-values, large values indicate a better reconstruction than smaller values.

The chosen test image  $u_{org}$ , shown in Figure 2.1(a), consists of squares of various sizes. We are interested in selecting  $\alpha_i$ ,  $i = 1, 2$ , such that the original image  $u_{org}$  is preserved

## I. Subspace Correction Methods for Total Variation Minimization

best. Therefore we compute the minimizer of the  $L^1$ - $L^2$ -TV model with  $g = u_{org}$  and  $T_i = I$ ,  $i = 1, 2$ , for  $\alpha_1, \alpha_2 \in \{0, 0.1, 0.2, \dots, 0.9, 1, 1.5\}$  and depict the obtained PSNR values in Figure 2.1(b). Note that for  $\alpha_1 = \alpha_2 = 0$  we set the PSNR value to the default value of 0. For  $\alpha_2 = 0$  and  $\alpha_1 > 0$  we see the typical behavior of the  $L^1$ -TV model. In fact, depending on the size of  $\alpha_1$ , different scales of the image features are preserved exactly. Typically, for decreasing  $\alpha_1$ , features at smaller scales are suddenly “lost” in the reconstruction, whereas other features are still recovered perfectly. Also the fading-away effect of image features at various scales depending on the decreasing choice of  $\alpha_2$  of the  $L^2$ -TV model can be seen clearly. However, Figure 2.1(b) shows that a combination of  $\alpha_1 > 0$ ,  $\alpha_2 > 0$  always gives a better restoration than setting one of the parameters to 0.

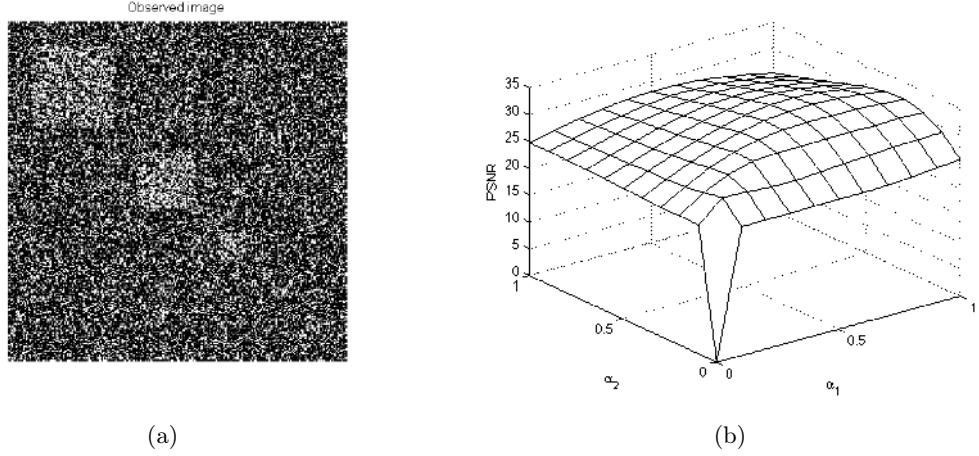


**Figure 2.1:** (a) Phantom image. (b) PSNR-values of the scale space generated by minimizing the  $L^1$ - $L^2$ -TV model for different choices of the parameters  $\alpha_1$  and  $\alpha_2$ .

In the second experiment we corrupt the original image of Figure 2.1(a) by *Gaussian noise and salt-and-pepper noise*; i.e.,  $g$  is now the image in Figure 2.2(a). Then we again compute the minimizer of the  $L^1$ - $L^2$ -TV model for  $\alpha_1, \alpha_2 \in \{0, 0.1, 0.2, \dots, 1\}$  and depict the obtained PSNR values in Figure 2.2(b). The maximal PSNR value is reached for  $\alpha_1 = 0.7$  and  $\alpha_2 = 0.4$ , which shows that for a combination of these two noise types the  $L^1$ - $L^2$ -TV model outperforms the  $L^1$ -TV model as well as the  $L^2$ -TV model.

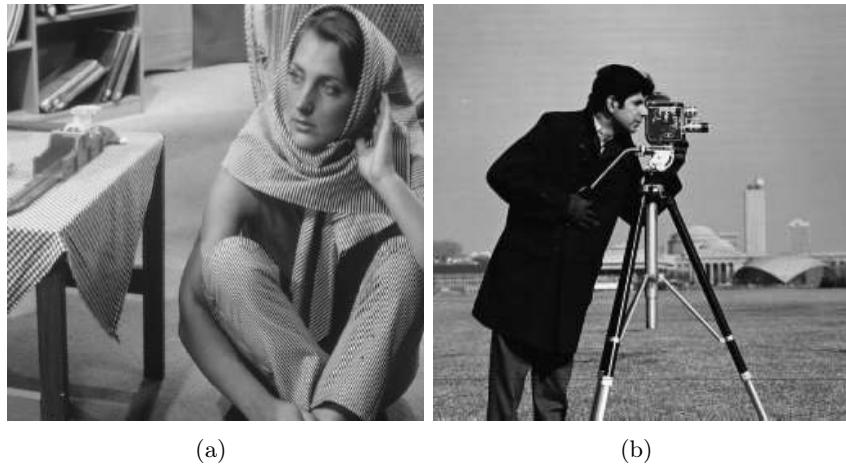
For the sake of a performance reference we also compare our  $L^1$ - $L^2$ -TV minimization algorithm with the frequently used ROAD-trilateral filter [31], which is designed to remove a mixture of Gaussian noise (with zero mean and variance  $\sigma$ ) and impulse noise. This filter is based on a simple statistic to detect outliers in an image. For our comparison we restore the Barbara image (see Figure 2.3(a)) and the Cameraman image (see Figure 2.3(b)) for salt-and-pepper noise with  $p_1 = p_2 \in \{0.05, 0.15\}$  and low levels of Gaussian noise, i.e.  $\sigma \in \{5/255, 15/255\}$ , as available in the literature. Here we further test the algorithms for higher levels of Gaussian noise, i.e.,  $\sigma \in \{\sqrt{0.02}, 0.2\}$ , plus salt-and-pepper noise; see Table 2.1. We also note that for impulse noise-dominated contamination of image data the implementation of strategies, such as the one in [14] and the references therein, enhances the performance of the algorithm. In the  $L^1$ - $L^2$ -TV minimization algorithm we set  $T_i = I$ ,  $i = 1, 2$ ,  $\alpha_1 \in \{0.1, 0.4, 0.7, 1, 1.3, 1.6, 2\}$ , and  $\alpha_2 \in \{0, 0.1, 0.4, 0.7, 1\}$ . For the ROAD-trilateral filter we choose  $\sigma_S = 1$ ,  $\sigma_I = 40/255$ ,

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems



**Figure 2.2:** (a) Image of Figure 2.1 corrupted by Gaussian noise (with zero mean and variance  $\sigma = 0.1$ ) and 75% salt-and-pepper noise (more precisely  $p_1 = 0.5$  and  $p_2 = 0.25$ ). (b) PSNR-values of the minimizer of the  $L^1-L^2$ -TV model for different choices of the parameters  $\alpha_1$  and  $\alpha_2$ .

$\sigma_J = 30/255$ , and  $\sigma_R$  is optimized between 10/255 and 50/255 as suggested in [23] with window-size  $3 \times 3$ . In Table 2.1 we show the highest PSNR-values achieved in our experiments for both methods. We observe that the  $L^1-L^2$ -TV minimization algorithm outperforms the ROAD-trilateral filter with respect to PSNR and that  $\alpha_2 = 0$  yields the best results in case of relatively high impulse noise.



**Figure 2.3:** (a) The original Barbara image of size  $512 \times 512$  pixels. (b) The original Cameraman image of size  $256 \times 256$  pixels.

## I. Subspace Correction Methods for Total Variation Minimization

	$\sigma$	$p_1 = p_2$	ROAD-trilateral	$L^1\text{-}L^2\text{-TV}$
Barbara	5/255	0.05	24.23	26.01 ( $\alpha_1 = 1.3, \alpha_2 = 1$ )
		0.15	22.03	24.56 ( $\alpha_1 = 1.3, \alpha_2 = 0$ )
	15/255	0.05	23.61	25.06 ( $\alpha_1 = 1.6, \alpha_2 = 0$ )
		0.15	21.62	23.78 ( $\alpha_1 = 1.3, \alpha_2 = 0$ )
	$\sqrt{0.02}$	0.05	22.19	23.18 ( $\alpha_1 = 0.7, \alpha_2 = 0.4$ )
		0.15	18.50	22.59 ( $\alpha_1 = 0.7, \alpha_2 = 0$ )
	$\sqrt{0.02}$	0.005	22.51	23.66 ( $\alpha_1 = 0.7, \alpha_2 = 1$ )
		0.01	22.49	23.60 ( $\alpha_1 = 0.7, \alpha_2 = 1$ )
	0.2	0.005	21.30	23.05 ( $\alpha_1 = 0.4, \alpha_2 = 1$ )
		0.01	21.26	23.05 ( $\alpha_1 = 0.4, \alpha_2 = 1$ )
Cameraman	5/255	0.05	23.96	27.12 ( $\alpha_1 = 1.3, \alpha_2 = 1$ )
		0.15	21.83	25.01 ( $\alpha_1 = 1.3, \alpha_2 = 0$ )
	15/255	0.05	23.72	25.92 ( $\alpha_1 = 1.3, \alpha_2 = 0.4$ )
		0.15	21.31	23.98 ( $\alpha_1 = 1.3, \alpha_2 = 0$ )
	$\sqrt{0.02}$	0.05	22.02	23.49 ( $\alpha_1 = 1, \alpha_2 = 0.1$ )
		0.15	18.45	22.33 ( $\alpha_1 = 1, \alpha_2 = 0$ )
	$\sqrt{0.02}$	0.005	22.48	24.23 ( $\alpha_1 = 0.7, \alpha_2 = 1$ )
		0.01	22.44	24.25 ( $\alpha_1 = 0.7, \alpha_2 = 1$ )
	0.2	0.005	22.13	23.29 ( $\alpha_1 = 0.7, \alpha_2 = 0.7$ )
		0.01	22.07	23.22 ( $\alpha_1 = 0.7, \alpha_2 = 0.7$ )

**Table 2.1:** PSNR results for the  $512 \times 512$  pixel image ‘‘Barbara’’ and the  $256 \times 256$  pixel image ‘‘Cameraman’’. The parameters of the ROAD-trilateral filter are  $\sigma_S = 1$ ,  $\sigma_I = 40/255$ ,  $\sigma_J = 30/255$ , and  $\sigma_R$  is optimized between  $10/255$  and  $50/255$  as suggested in [23]. For the  $L^1\text{-}L^2\text{-TV}$  minimization algorithm we show in parentheses the parameters for which the best PSNR is obtained.

### 2.3. Subspace Correction Approach in $L^2(\Omega)$

In order to enable or to accelerate the solution process, subspace correction and domain decomposition methods offer the potential to split the computational workload and to solve (in parallel) a sequence of more tractable problems. In this sense we now follow the general philosophy of subspace correction and seek to minimize  $J_{\alpha_1, \alpha_2}$  by decomposing  $L^2(\Omega)$  into two appropriate subspaces  $V_1$  and  $V_2$  such that  $L^2(\Omega) = V_1 + V_2$ . With this splitting we aim to solve (2.2) by the following alternating algorithm.

Choose an initial  $u^{(0)} =: \tilde{u}_1^{(0)} + \tilde{u}_2^{(0)} \in V_1 + V_2$ , for example,  $u^{(0)} = 0$ , and iterate

$$\left\{ \begin{array}{l} u_1^{(n+1)} \leftarrow \arg \min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(n)}), \\ u_2^{(n+1)} \leftarrow \arg \min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + u_2), \\ u^{(n+1)} \leftarrow u_1^{(n+1)} + u_2^{(n+1)}, \\ \tilde{u}_1^{(n+1)} \leftarrow \chi_1 \cdot u^{(n+1)}, \\ \tilde{u}_2^{(n+1)} \leftarrow \chi_2 \cdot u^{(n+1)}, \end{array} \right. \quad (2.9)$$

where  $\chi_1, \chi_2 \in L^\infty(\Omega)$  have the properties (a)  $\chi_1 + \chi_2 = 1$  and (b)  $\chi_i \in V_i$  for  $i = 1, 2$ . Let  $\kappa := \max\{\|\chi_1\|_\infty, \|\chi_2\|_\infty\} < \infty$ . Although  $\tilde{u}_1^{(n+1)}$  is essentially not used in the above algorithm, it is present for theoretical reasons. From the assumptions on  $\chi_i$  we obtain that

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

$u^{(n)} = (\chi_1 + \chi_2)u^{(n)} = \tilde{u}_1^{(n)} + \tilde{u}_2^{(n)}$ . Further, if the  $V_i$ 's are orthogonal, i.e.,  $L^2(\Omega) = V_1 \oplus V_2$ , then  $\tilde{u}_i^{(n)} = u_i^{(n)}$  for all  $n \in \mathbb{N}$ , and hence in this case there is no need to introduce the variables  $\tilde{u}_i^{(n)}$ ; cf. (2.48) below. The parallel version of the algorithm in (2.9) reads as follows.

Choose an initial  $u^{(0)} =: \tilde{u}_1^{(0)} + \tilde{u}_2^{(0)} \in V_1 + V_2$ , for example  $u^{(0)} = 0$ , and iterate

$$\begin{cases} u_1^{(n+1)} \leftarrow \arg \min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(n)}), \\ u_2^{(n+1)} \leftarrow \arg \min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(n)} + u_2), \\ u^{(n+1)} \leftarrow \frac{\tilde{u}_1^{(n+1)} + u_2^{(n+1)} + u^{(n)}}{2}, \\ \tilde{u}_1^{(n+1)} \leftarrow \chi_1 \cdot u^{(n+1)}, \\ \tilde{u}_2^{(n+1)} \leftarrow \chi_2 \cdot u^{(n+1)}. \end{cases} \quad (2.10)$$

We define the orthogonal complement of  $V_i$  in  $L^2(\Omega)$  by  $V_i^c$ , i.e.,  $L^2(\Omega) = V_i \oplus V_i^c$ , and we define by  $\pi_{V_i}$  the corresponding orthogonal projection onto  $V_i$ . Moreover, we define the domain of a functional  $\mathcal{J} : L^2(\Omega) \rightarrow \bar{\mathbb{R}}$  as the set  $\text{Dom}(\mathcal{J}) = \{v \in L^2(\Omega) : \mathcal{J}(v) \neq \infty\}$ .

Note that the subspace minimization problems in (2.9) and (2.10) can be written as constrained optimization problems of the form

$$\min_{v \in L^2(\Omega)} J_{\alpha_1, \alpha_2}(v) \quad \text{subject to (s.t.) } Av = b,$$

where  $A : L^2(\Omega) \rightarrow L^2(\Omega)$  is a linear and continuous operator on  $L^2(\Omega)$  and  $b \in L^2(\Omega)$ . In particular, we have

$$\min_{v \in L^2(\Omega)} J_{\alpha_1, \alpha_2}(v + b) \quad \text{s.t. } \pi_{V_i^c} v = 0,$$

or equivalently

$$\min_{v \in L^2(\Omega)} J_{\alpha_1, \alpha_2}(v) \quad \text{s.t. } \pi_{V_i^c}(v) = \pi_{V_i^c}(b), \quad (2.11)$$

where  $b = u_1^{(n+1)}$  for the second minimization problem in (2.9) and  $b = \tilde{u}_j^{(n)}$  for the first minimization problem in (2.9) and the minimization problems in (2.10) for  $i = 1, 2$  and  $j \in \{1, 2\} \setminus \{i\}$ .

For any *attainable*  $b \in V_j$ , i.e., when there exists an  $u \in \text{Dom}(J_{\alpha_1, \alpha_2})$  such that  $\pi_{V_i^c}(u) = \pi_{V_i^c}(b)$ , we observe that  $\{u \in L^2(\Omega) : \pi_{V_i^c}(u) = \pi_{V_i^c}(b), J_{\alpha_1, \alpha_2}(u) \leq C\} \subset \{J_{\alpha_1, \alpha_2} \leq C\}$  for all  $C > 0$ ,  $i = 1, 2$ , and  $j \in \{1, 2\} \setminus \{i\}$ . Hence the former set is bounded by the coercivity assumption, and thus (2.11) has a solution, since every  $u_i^{(n)}$  and  $\tilde{u}_i^{(n)}$  generated by the algorithm in (2.9) and (2.10) is attainable.

**Proposition 2.2.** *The algorithms in (2.9) and (2.10) produce a sequence  $(u^{(n)})_n$  in  $L^2(\Omega)$  with the following properties:*

- (i)  $J_{\alpha_1, \alpha_2}(u^{(n)}) \geq J_{\alpha_1, \alpha_2}(u^{(n+1)})$  for all  $n \in \mathbb{N}$ ;
- (ii) the sequence  $(u^{(n)})_n$  has subsequences that weakly converge in  $L^2(\Omega)$  and  $BV(\Omega)$ .

*Proof.* First we show (i) for the algorithm in (2.9). Observe that

$$\begin{aligned} J_{\alpha_1, \alpha_2}(u^{(n)}) &= J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(n)} + \tilde{u}_2^{(n)}) \geq J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + \tilde{u}_2^{(n)}) \geq J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + u_2^{(n+1)}) \\ &= J_{\alpha_1, \alpha_2}(u^{(n+1)}), \end{aligned}$$

## I. Subspace Correction Methods for Total Variation Minimization

which proves the assertion.

To show (i) for the algorithm in (2.10) we consider first that

$$J_{\alpha_1, \alpha_2}(u^{(n)}) \geq \frac{1}{2} \left( J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + \tilde{u}_2^{(n)}) + J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(n)} + u_2^{(n+1)}) \right).$$

Moreover, by convexity we obtain

$$J_{\alpha_1, \alpha_2} \left( \frac{u_1^{(n+1)} + u_2^{(n+1)} + \tilde{u}_1^{(n)} + \tilde{u}_2^{(n)}}{2} \right) \leq \frac{1}{2} \left( J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + \tilde{u}_2^{(n)}) + J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(n)} + u_2^{(n+1)}) \right)$$

and hence  $J_{\alpha_1, \alpha_2}(u^{(n)}) \geq J_{\alpha_1, \alpha_2}(u^{(n+1)})$ .

From the above considerations we infer that  $J_{\alpha_1, \alpha_2}(u^{(0)}) \geq J_{\alpha_1, \alpha_2}(u^{(n)})$  for all  $n \in \mathbb{N}$ . By the coercivity condition on  $J_{\alpha_1, \alpha_2}$ ,  $(u^{(n)})_n$  is uniformly bounded in  $L^2(\Omega)$ , and hence there exists a weakly convergent subsequence. Moreover, due to the presence of  $\varphi(|Du|)$  in  $J_{\alpha_1, \alpha_2}$  and  $\alpha_1 + \alpha_2 > 0$  we obtain that  $(u^{(n)})_n$  is bounded in  $BV(\Omega)$ . The compact embedding  $BV(\Omega) \hookrightarrow L^q(\Omega)$ ,  $q < \frac{d}{d-1}$ , implies that a subsequence  $(u^{(n_k)})_k$  converges in  $L^q(\Omega)$  to a limit  $u^{(\infty)} \in L^2(\Omega)$ . By [3, Prop. 10.1.1] we even have that  $u^{(\infty)} \in BV(\Omega)$ ,  $\liminf_{n \rightarrow \infty} \varphi(|Du^{(n_k)}|)(\Omega) \geq \varphi(|Du^{(\infty)}|)(\Omega)$ , and  $u^{(n_k)}$  weakly converges to  $u^{(\infty)}$  in  $BV(\Omega)$ , which concludes the proof.  $\square$

**Remark 2.3.** Since the sequence  $(J_{\alpha_1, \alpha_2}(u^{(n)}))_n$  is monotonically decreasing and bounded from below, it is also convergent.

**Proposition 2.4.** The sequences  $(u_i^{(n)})_n$  and  $(\tilde{u}_i^{(n)})_n$  for  $i = 1, 2$  generated by the algorithms in (2.9) or (2.10) are bounded in  $L^2(\Omega)$  and hence have weak accumulation points  $u_i^{(\infty)} \in L^2(\Omega)$  and  $\tilde{u}_i^{(\infty)} \in L^2(\Omega)$ , respectively.

*Proof.* The boundedness of  $(u^{(n)})_n$  implies the boundedness of  $(\tilde{u}_i^{(n)})_n$ , since

$$\|\tilde{u}_i^{(n)}\|_{L^2(\Omega)} = \|\chi_i u^{(n)}\|_{L^2(\Omega)} \leq \kappa \|u^{(n)}\|_{L^2(\Omega)} \leq C < \infty \quad \text{for } i = 1, 2, \quad (2.12)$$

where  $\kappa$  is defined as below (2.9). By the definition of  $u_1^{(n+1)}$  and by the coercivity assumption on  $J_{\alpha_1, \alpha_2}$  we have that  $(u_1^{(n+1)} + \tilde{u}_2^{(n)})_n$  is bounded in  $L^2(\Omega)$ ; i.e., there exists a constant  $C > 0$  such that  $\|u_1^{(n+1)} + \tilde{u}_2^{(n)}\|_{L^2(\Omega)} \leq C$  for all  $n \in \mathbb{N}$ . Since  $(\tilde{u}_2^{(n)})_n$  is bounded in  $L^2(\Omega)$  by (2.12), the triangle inequality yields

$$\|u_1^{(n+1)}\|_{L^2(\Omega)} - \|\tilde{u}_2^{(n)}\|_{L^2(\Omega)} \leq \|u_1^{(n+1)} + \tilde{u}_2^{(n)}\|_{L^2(\Omega)} \leq C.$$

Hence  $(u_1^{(n)})_n$  is bounded in  $L^2(\Omega)$ . By similar arguments we get the  $L^2(\Omega)$ -boundedness of  $(u_2^{(n)})_n$ . Consequently  $(\tilde{u}_i^{(n)})_n$  and  $(u_i^{(n)})_n$  have a weakly convergent subsequence with limits  $\tilde{u}_i^{(\infty)}$  and  $u_i^{(\infty)}$ , respectively, in  $L^2(\Omega)$ .  $\square$

## 2.4. Subspace Correction Approach for the Discretized Problem

In this section we analyze the method as it is implemented in finite dimensions upon discretization. We note, however, that a function space analysis appears possible as well, but would require careful handling of weak and weak-\* convergent sequences as well as properties of the convex subdifferential.

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

### 2.4.1. Notation and Basic Definitions

In the rest of the paper we work on a finite regular mesh as a discretization of  $\Omega$ . We approximate functions  $u$  by discrete functions, again denoted by  $u$  with  $\nabla u$  representing their gradient. Instead of the continuous functional (2.2) we consider its discrete approximation, for ease again denoted by  $J_{\alpha_1, \alpha_2}$  in (2.13) below. Note that the discrete approximation  $\Gamma$ -converges to the continuous functional in (2.2) (see [7, 37]), and it has the same structural properties as the continuous one. Although in our applications we are mainly interested in imaging problems, i.e., two-dimensional problems, our notation covers any  $d$ -dimensional space.

In our discrete setting we define the *discrete  $d$ -orthotope*  $\Omega = \{x_1^1 < \dots < x_{N_1}^1\} \times \dots \times \{x_1^d < \dots < x_{N_d}^d\} \subset \mathbb{R}^d$ ,  $d \in \mathbb{N}$ , and the underlying “function space” is  $\mathcal{H} = \mathbb{R}^{N_1 \times N_2 \times \dots \times N_d}$ , where  $N_j \in \mathbb{N}$  for  $j = 1, \dots, d$ . Accordingly,  $V_1$  and  $V_2$  are appropriate subspaces of  $\mathcal{H}$  such that  $\mathcal{H} = V_1 + V_2$ , and by  $V_i^c$  we denote the orthogonal complement of  $V_i$  in  $\mathcal{H}$  for  $i = 1, 2$ . For  $u \in \mathcal{H}$  we write  $u = u(x) = u(x_{i_1}^1, \dots, x_{i_d}^d)$ , where  $i_j \in \{1, \dots, N_j\}$  and  $x \in \Omega$ . Let  $h = x_{i_j+1}^j - x_{i_j}^j$  be the equidistant step-size for all  $j = 1, \dots, d$ . We define the scalar products of  $u, v \in \mathcal{H}$  and of  $p, q \in \mathcal{H}^d$  by

$$\langle u, v \rangle_{\mathcal{H}} = h^d \sum_{x \in \Omega} u(x)v(x) \quad \text{and} \quad \langle p, q \rangle_{\mathcal{H}^d} = h^d \sum_{x \in \Omega} \langle p(x), q(x) \rangle_{\mathbb{R}^d}$$

with  $\langle y, z \rangle_{\mathbb{R}^d} = \sum_{j=1}^d y_j z_j$  for every  $y = (y_1, \dots, y_d) \in \mathbb{R}^d$  and  $z = (z_1, \dots, z_d) \in \mathbb{R}^d$ . In what follows we consider different norms. In particular we use

$$\|u\|_{\ell^p(\Omega)} = \left( h^d \sum_{x \in \Omega} |u(x)|^p \right)^{1/p}, \quad 1 \leq p < \infty,$$

and  $\|u\|_{\ell^\infty(\Omega)} = \sup_{x \in \Omega} |u(x)|$ . Sometimes we do not specify the norm; i.e., we just write  $\|\cdot\|$ , which indicates that any norm can be taken.

The discrete gradient  $\nabla u$  is denoted by  $(\nabla u)(x) = ((\nabla u)^1(x), \dots, (\nabla u)^d(x))$  with

$$(\nabla u)^j(x) = \frac{1}{h} \cdot \begin{cases} u(x_{i_1}^1, \dots, x_{i_j+1}^j, \dots, x_{i_d}^d) - u(x_{i_1}^1, \dots, x_{i_j}^j, \dots, x_{i_d}^d) & \text{if } i_j < N_j, \\ 0 & \text{if } i_j = N_j, \end{cases}$$

for all  $j = 1, \dots, d$  and for all  $x \in \Omega$ . Let  $\varphi : \mathbb{R} \rightarrow \mathbb{R}$  with

$$\varphi(|\omega|)(\Omega) := h^d \sum_{x \in \Omega} \varphi(|\omega(x)|),$$

for  $\omega \in \mathcal{H}^d$ , where  $|y| = \sqrt{y_1^2 + \dots + y_d^2}$ . In particular we define the *total variation* of  $u$  by setting  $\varphi(t) = t$  and  $\omega = \nabla u$ ; i.e.,

$$|\nabla u|(\Omega) := h^d \sum_{x \in \Omega} |\nabla u(x)|.$$

For an operator  $T$  we denote by  $T^*$  its adjoint. Further we introduce the *discrete divergence*  $\operatorname{div} : \mathcal{H}^d \rightarrow \mathcal{H}$  defined by  $\operatorname{div} = -\nabla^*$  ( $\nabla^*$  is the adjoint of the gradient  $\nabla$ ), in analogy to the

## I. Subspace Correction Methods for Total Variation Minimization

continuous setting. In our case, the discrete divergence operator is explicitly given by

$$(\operatorname{div} p)(x) = \frac{1}{h} \cdot \begin{cases} (p^1(x_{i_1}^1, \dots, x_{i_d}^d) - p^1(x_{i_1-1}^1, \dots, x_{i_d}^d)) & \text{if } 1 < i_1 < N_1, \\ p^1(x_{i_1}^1, \dots, x_{i_d}^d) & \text{if } i_1 = 1, \\ -p^1(x_{i_1-1}^1, \dots, x_{i_d}^d) & \text{if } i_1 = N_1, \end{cases} + \dots + \frac{1}{h} \cdot \begin{cases} (p^d(x_{i_1}^1, \dots, x_{i_d}^d) - p^d(x_{i_1}^1, \dots, x_{i_d-1}^d)) & \text{if } 1 < i_d < N_d, \\ p^d(x_{i_1}^1, \dots, x_{i_d}^d) & \text{if } i_d = 1, \\ -p^d(x_{i_1}^1, \dots, x_{i_d-1}^d) & \text{if } i_d = N_d, \end{cases}$$

for every  $p = (p^1, \dots, p^d) \in \mathcal{H}^d$  and for all  $x \in \Omega$ . (Note that if the discrete domains  $\Omega$  are not discrete  $d$ -orthotopes, then the definitions of the gradient and divergence operators have to be adjusted accordingly.) We will often use the symbol  $1$  to indicate the constant vector with entry values  $1$ , and  $1_D$  to indicate the characteristic function of the domain  $D \subset \Omega$ .

For a convex functional  $\mathcal{J} : \mathcal{H} \rightarrow \bar{\mathbb{R}}$ , we define the *subdifferential* of  $\mathcal{J}$  at  $v \in \mathcal{H}$  as the set valued mapping

$$\partial\mathcal{J}(v) := \begin{cases} \emptyset & \text{if } \mathcal{J}(v) = \infty, \\ \{v^* \in \mathcal{H} : \langle v^*, u - v \rangle_{\mathcal{H}} + \mathcal{J}(v) \leq \mathcal{J}(u) \quad \forall u \in \mathcal{H}\} & \text{otherwise} \end{cases}$$

at  $v \in \mathcal{H}$ . It is clear from this definition that  $0 \in \partial\mathcal{J}(v)$  if and only if  $v$  is a minimizer of  $\mathcal{J}$ . Since we consider different spaces, namely  $\mathcal{H}$ ,  $V_i$ , it is sometimes useful to indicate this in the notation of the subdifferential; e.g., we write  $\partial_{V_i}\mathcal{J}$  whenever the subdifferential of  $\mathcal{J}$  is taken with respect to  $V_i$ .

### 2.4.2. Properties of Subspace Correction Methods

In what follows we consider the discrete functional

$$J_{\alpha_1, \alpha_2}(u) = \alpha_1 \|T_1 u - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{\ell^2(\Omega)}^2 + \varphi(|\nabla u|)(\Omega), \quad (2.13)$$

where  $T_i : \mathcal{H} \rightarrow \mathcal{H}$  is a bounded linear operator,  $g_i \in \mathcal{H}$  is a given datum, and  $\alpha_i \geq 0$  for  $i = 1, 2$  with  $\alpha_1 + \alpha_2 > 0$ . Moreover, we assume that  $\varphi$  fulfills assumption  $(A_\varphi)$  and that  $J_{\alpha_1, \alpha_2}$  is bounded from below and coercive.

In the sequential and parallel algorithms in (2.9) and (2.10) we denote the difference between the current subspace minimizer  $u_i^{(n+1)}$  and the initial value  $\tilde{u}_i^{(n)}$  by  $s^{(n+\frac{i}{2})}$ ; i.e.,

$$s^{(n+\frac{i}{2})} := u_i^{(n+1)} - \tilde{u}_i^{(n)} \quad \text{for } i = 1, 2. \quad (2.14)$$

It is easy to see that in the sequential domain decomposition algorithm

$$s^{(n+\frac{i}{2})} = \arg \min_s J_{\alpha_1, \alpha_2}(u^{(n)} + (i-1)s^{(n+\frac{i-1}{2})} + s) \quad \text{s.t. } \pi_{V_i^c}s = 0,$$

and in the parallel version

$$s^{(n+\frac{i}{2})} = \arg \min_s J_{\alpha_1, \alpha_2}(u^{(n)} + s) \quad \text{s.t. } \pi_{V_i^c}s = 0$$

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

for  $i = 1, 2$ . Moreover, for  $v, s \in \mathcal{H}$  a quadratic Taylor expansion yields

$$\begin{aligned} \min_s J_{\alpha_1, \alpha_2}(v + s) &= \min_s 2\alpha_2 \langle T_2 s, T_2 v - g_2 \rangle_{\mathcal{H}} + \alpha_2 \|T_2 s\|_{\ell^2(\Omega)}^2 \\ &\quad + \alpha_1 \|T_1(v + s) - g_1\|_{\ell^1(\Omega)} + \varphi(|\nabla(v + s)|)(\Omega). \end{aligned}$$

Then the following lemma can be proven similarly to Lemma 1 of [49]. For its statement we define the quantities  $v(i)$ ,  $i = 1, 2$ , as follows: For the sequential domain decomposition algorithm in (2.9) choose  $v(i) = u^{(n)}$  if  $i = 1$  and  $v(i) = u^{(n)} + s^{(n+\frac{i}{2})}$  for  $i = 2$ , while for the parallel domain decomposition algorithm in (2.10),  $v(i) = u^{(n)}$  for  $i = 1, 2$ .

**Lemma 2.5.** *Let  $P(u) = \alpha_1 \|T_1 u - g_1\|_{\ell^1(\Omega)} + \varphi(|\nabla u|)(\Omega)$ . For any  $v(i) \in \mathcal{H}$  chosen according to the underlying algorithm, let  $\tilde{s} = s^{(n+\frac{i}{2})}$  for  $i = 1, 2$ . Then*

$$J_{\alpha_1, \alpha_2}(v(i) + \tilde{s}) = J_{\alpha_1, \alpha_2}(v(i)) + 2\alpha_2 \langle T_2 \tilde{s}, T_2 v(i) - g_2 \rangle_{\mathcal{H}} + \alpha_2 \|T_2 \tilde{s}\|_{\ell^2(\Omega)}^2 + P(v(i) + \tilde{s}) - P(v(i))$$

and

$$2\alpha_2 \langle T_2 \tilde{s}, T_2 v(i) - g_2 \rangle_{\mathcal{H}} + P(v(i) + \tilde{s}) - P(v(i)) \leq -2\alpha_2 \|T_2 \tilde{s}\|_{\ell^2(\Omega)}^2.$$

**Remark 2.6.** *With  $\alpha_2 > 0$ ,  $v(i)$  as above and  $\tilde{s} = s^{(n+\frac{i}{2})}$ , a direct consequence of Lemma 2.5 is that*

$$J_{\alpha_1, \alpha_2}(v(i) + \tilde{s}) - J_{\alpha_1, \alpha_2}(v(i)) \leq -\alpha_2 \|T_2 \tilde{s}\|_{\ell^2(\Omega)}^2, \quad (2.15)$$

where  $\alpha_2 \|T_2 \tilde{s}\|_{\ell^2(\Omega)}^2 > 0$  whenever  $\tilde{s} \notin \ker T_2$ . Note that the above descent property holds in particular when  $T_2^* T_2$  is invertible and  $\|\tilde{s}\| \neq 0$ .

**Proposition 2.7.** *Assume that  $T_2^* T_2$  is invertible and  $\alpha_2 > 0$ . Let the sequence  $(u^{(n)})_n$  be generated by the algorithm in (2.9) or (2.10), and let  $s^{(n+\frac{i}{2})}$  be defined as in (2.14). Then we have the following statements:*

- (i)  $\|s^{(n+\frac{i}{2})}\| \rightarrow 0$  for  $n \rightarrow \infty$ ,
- (ii)  $\|u^{(n+1)} - u^{(n)}\| \rightarrow 0$  for  $n \rightarrow \infty$ .

*Proof.* We begin by showing these statements for the sequential algorithm in (2.9). By the minimality property of  $s^{(n+\frac{i}{2})}$  we have that whenever  $\|s^{(n+\frac{i}{2})}\| \neq 0$ ,

$$J_{\alpha_1, \alpha_2}(u^{(n)} + (i-1)s^{(n+\frac{i-1}{2})} + s^{(n+\frac{i}{2})}) < J_{\alpha_1, \alpha_2}(u^{(n)} + (i-1)s^{(n+\frac{i-1}{2})})$$

for  $i = 1, 2$ . Note that  $u^{(n+1)} = u^{(n)} + s^{(n+\frac{1}{2})} + s^{(n+1)}$ . Analogously to Remark 2.3, we find that  $J_{\alpha_1, \alpha_2}$  is convergent, and hence by the above observation we obtain

$$J_{\alpha_1, \alpha_2}(u^{(n)} + (i-1)s^{(n+\frac{i-1}{2})} + s^{(n+\frac{i}{2})}) - J_{\alpha_1, \alpha_2}(u^{(n)} + (i-1)s^{(n+\frac{i-1}{2})}) \rightarrow 0 \quad \text{for } n \rightarrow \infty.$$

By (2.15) it follows then that  $\|s^{(n+\frac{i}{2})}\| \rightarrow 0$  for  $n \rightarrow \infty$  and for  $i = 1, 2$ , which proves (i). Since  $u^{(n+1)} = u^{(n)} + s^{(n+\frac{1}{2})} + s^{(n+1)}$ , (ii) immediately follows.

For the parallel algorithm in (2.10) we obtain by the minimality property of  $s^{(n+\frac{i}{2})}$  that

$$J_{\alpha_1, \alpha_2}(u^{(n)} + s^{(n+\frac{i}{2})}) < J_{\alpha_1, \alpha_2}(u^{(n)}) \quad (2.16)$$

## I. Subspace Correction Methods for Total Variation Minimization

for  $i = 1, 2$  whenever  $\|s^{(n+\frac{i}{2})}\| \neq 0$ . Hence by convexity and the definition of  $u^{(n+1)}$  in (2.10) we get

$$2J_{\alpha_1, \alpha_2}(u^{(n)}) > J_{\alpha_1, \alpha_2}(u^{(n)} + s^{(n+\frac{1}{2})}) + J_{\alpha_1, \alpha_2}(u^{(n)} + s^{(n+1)}) \geq 2J_{\alpha_1, \alpha_2}(u^{(n+1)}).$$

From (2.16), the convergence of  $J_{\alpha_1, \alpha_2}$ , and the previous inequalities we obtain

$$\begin{aligned} & \underbrace{J_{\alpha_1, \alpha_2}(u^{(n)}) - J_{\alpha_1, \alpha_2}(u^{(n)} + s^{(n+\frac{1}{2})})}_{\geq 0} + \underbrace{J_{\alpha_1, \alpha_2}(u^{(n)}) - J_{\alpha_1, \alpha_2}(u^{(n)} + s^{(n+1)})}_{\geq 0} \\ & \leq 2(J_{\alpha_1, \alpha_2}(u^{(n)}) - J_{\alpha_1, \alpha_2}(u^{(n+1)})) \rightarrow 0 \quad \text{for } n \rightarrow \infty. \end{aligned}$$

By (2.15) we eventually have that  $\|s^{(n+\frac{i}{2})}\| \rightarrow 0$  for  $n \rightarrow \infty$  and for  $i = 1, 2$ . The second statement follows from  $u^{(n+1)} = \frac{u_1^{(n+1)} + u_2^{(n+1)} + u^{(n)}}{2}$ , since

$$\|u^{(n+1)} - u^{(n)}\| = \left\| \frac{u_1^{(n+1)} + u_2^{(n+1)} - u^{(n)}}{2} \right\| = \left\| \frac{s^{(n+\frac{1}{2})} + s^{(n+1)}}{2} \right\|.$$

□

**Remark 2.8.** From the proof of Proposition 2.7 we find that if  $T_2^* T_2$  is invertible and  $\alpha_2 > 0$ , then we can replace the nonincrease of the energy  $J_{\alpha_1, \alpha_2}$  in Proposition 2.2(i) by a strict monotone decrease unless  $u^{(n+1)} = u^{(n)}$ .

### 2.4.2.1. A Convergence Estimate

For proving convergence results of the algorithms in (2.9) and (2.10) we use a characterization of solutions of the minimization problem (2.2) similar to [50, Prop. 4.1] in the continuous setting for  $\alpha_1 = 0$  and adapted in [29, Prop. 5.2] to the discrete case.

**Characterization of Solutions** Following [26, Def. 4.1, p. 17] the *conjugate (or Legendre transform)* of a convex function  $\phi : V \rightarrow \mathbb{R}$ , with  $V$  a vector space with topological dual  $V^*$  and duality pairing  $\langle \cdot, \cdot \rangle$ , is defined by

$$\phi^*(u^*) = \sup_{u \in V} \{\langle u, u^* \rangle - \phi(u)\}.$$

The convex conjugate is useful when characterizing the solution to (2.13) with  $\alpha_1, \alpha_2 > 0$ .

**Proposition 2.9.** Let  $\zeta, u \in \mathcal{H}$ . If assumption  $(A_\varphi)$  holds true, then  $\zeta \in \partial J_{\alpha_1, \alpha_2}(u)$  if and only if there exist  $M = (M_0, M_1, M_2) \in \mathcal{H}^d \times \mathcal{H} \times \mathcal{H}$  and a constant  $c_1 \geq 0$  with  $|M_0(x)| \leq c_1$ ,  $|M_1(x)| \leq \alpha_1$  for all  $x \in \Omega$  such that

$$\varphi(|(\nabla u)(x)|) + \langle M_0(x), \nabla u(x) \rangle_{\mathbb{R}^d} + \varphi_1^*(|M_0(x)|) = 0 \quad \forall x \in \Omega, \quad (2.17)$$

$$M_2(x) = -2\alpha_2(T_2 u - g_2)(x) \quad \forall x \in \Omega, \quad (2.18)$$

$$\alpha_1 |(T_1 u - g_1)(x)| + M_1(x)((T_1 u)(x) - g_1(x)) = 0 \quad \forall x \in \Omega, \quad (2.19)$$

$$T_1^* M_1 + T_2^* M_2 - \operatorname{div} M_0 + \zeta = 0, \quad (2.20)$$

where  $\varphi_1^*$  is the conjugate function of  $\varphi_1$  defined by  $\varphi_1(t) = \varphi(|t|)$  for  $t \in \mathbb{R}$ . If, additionally,  $\varphi$  is differentiable and  $|(\nabla u)(x)| \neq 0$  for  $x \in \Omega$ , then we can compute  $M_0$  as

$$M_0(x) = -\frac{\varphi'(|(\nabla u)(x)|)}{|(\nabla u)(x)|}(\nabla u)(x). \quad (2.21)$$

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

*Proof.* The proof of Proposition 2.9 as stated here is similar to that in [50, Prop. 4.1]. It is clear that  $\zeta \in \partial J_{\alpha_1, \alpha_2}(u)$  if and only if  $u \in \operatorname{argmin}_{v \in \mathcal{H}} \{J_{\alpha_1, \alpha_2}(v) - \langle \zeta, v \rangle_{\mathcal{H}}\}$ ; let us consider the following variational problem:

$$\inf_{v \in \mathcal{H}} \{J_{\alpha_1, \alpha_2}(v) - \langle \zeta, v \rangle_{\mathcal{H}}\} = \inf_{v \in \mathcal{H}} \{\alpha_1 \|T_1 v - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 v - g_2\|_{\ell^2(\Omega)}^2 + \varphi(|\nabla v|)(\Omega) - \langle \zeta, v \rangle_{\mathcal{H}}\}. \quad (\mathcal{P})$$

We denote such an infimum by  $\inf(\mathcal{P})$ . Now we compute  $(\mathcal{P}^*)$ , the dual of  $(\mathcal{P})$ . Let  $\mathcal{F} : \mathcal{H} \rightarrow \mathbb{R}$ ,  $\mathcal{G} : \mathcal{H}^d \times \mathcal{H} \times \mathcal{H} \rightarrow \mathbb{R}$ ,  $\mathcal{G}_0 : \mathcal{H}^d \rightarrow \mathbb{R}$ ,  $\mathcal{G}_1 : \mathcal{H} \rightarrow \mathbb{R}$ ,  $\mathcal{G}_2 : \mathcal{H} \rightarrow \mathbb{R}$  such that  $\mathcal{F}(v) = -\langle \zeta, v \rangle_{\mathcal{H}}$ ,  $\mathcal{G}_0(w_0) = \varphi(|w_0|)(\Omega)$ ,  $\mathcal{G}_1(\bar{w}) = \alpha_1 \|w_1 - g_1\|_{\ell^1(\Omega)}$ ,  $\mathcal{G}_2(\bar{w}) = \alpha_2 \|w_2 - g_2\|_{\ell^2(\Omega)}^2$ ,  $\mathcal{G}(w) = \mathcal{G}_0(w_0) + \mathcal{G}_1(w_1) + \mathcal{G}_2(w_2)$ , with  $w = (w_0, w_1, w_2) \in \mathcal{H}^d \times \mathcal{H} \times \mathcal{H}$ . Then the dual problem of  $(\mathcal{P})$  is given by (cf. [26, p. 60])

$$\sup_{p^* \in \mathcal{H}^d \times \mathcal{H} \times \mathcal{H}} \{-\mathcal{F}^*(\Lambda^* p^*) - \mathcal{G}^*(-p^*)\}, \quad (\mathcal{P}^*)$$

where  $\Lambda : \mathcal{H} \rightarrow \mathcal{H}^d \times \mathcal{H} \times \mathcal{H}$  is defined by  $\Lambda v = ((\nabla v)^1, \dots, (\nabla v)^d, T_1 v, T_2 v)$  and  $\Lambda^*$  is its adjoint. We denote the supremum in  $(\mathcal{P}^*)$  by  $\sup(\mathcal{P}^*)$ . Using the definition of the conjugate function, we compute  $\mathcal{F}^*$  and  $\mathcal{G}^*$ . In particular, we have

$$\mathcal{F}^*(\Lambda^* p^*) = \sup_{v \in \mathcal{H}} \{\langle \Lambda^* p^*, v \rangle_{\mathcal{H}} - \mathcal{F}(v)\} = \sup_{v \in \mathcal{H}} \langle \Lambda^* p^* + \zeta, v \rangle_{\mathcal{H}} = \begin{cases} 0 & \text{if } \Lambda^* p^* + \zeta = 0, \\ \infty & \text{otherwise,} \end{cases}$$

where  $p^* = (p_0^*, p_1^*, p_2^*)$ , and due to the separability of  $\mathcal{G}$  we find

$$\begin{aligned} \mathcal{G}^*(p^*) &= \sup_{w \in \mathcal{H}^d \times \mathcal{H} \times \mathcal{H}} \{\langle p^*, w \rangle_{\mathcal{H}^d \times \mathcal{H} \times \mathcal{H}} - \mathcal{G}(w)\} \\ &= \sup_{w_0 \in \mathcal{H}} \{\langle p_0^*, w_0 \rangle_{\mathcal{H}^d} - \mathcal{G}_0(w_0)\} + \sup_{w_1 \in \mathcal{H}} \{\langle p_1^*, w_1 \rangle_{\mathcal{H}} - \mathcal{G}_1(w_1)\} + \sup_{w_2 \in \mathcal{H}} \{\langle p_2^*, w_2 \rangle_{\mathcal{H}} - \mathcal{G}_2(w_2)\} \\ &= \mathcal{G}_0^*(p_0^*) + \mathcal{G}_1^*(p_1^*) + \mathcal{G}_2^*(p_2^*). \end{aligned}$$

We have that  $\mathcal{G}_2^*(p_2^*) = \langle \frac{p_2^*}{4\alpha_2} + g_2, p_2^* \rangle_{\mathcal{H}}$ ,  $\mathcal{G}_1^*(p_1^*) = \langle p_1^*, g_1 \rangle_{\mathcal{H}}$  if  $|p_1^*| \leq \alpha_1$ , and (see [26])  $\mathcal{G}_0^*(p_0^*) = \varphi_+^*(|p_0^*|)(\Omega)$  if  $|p_0^*(x)| \in \operatorname{Dom} \varphi_+$ , where  $\varphi_+$  is the conjugate function of  $\varphi_+$  defined by  $\varphi_+(t) := \varphi(|t|)$  for  $t \in \mathbb{R}$ . Therefore we can write  $(\mathcal{P}^*)$  in the following way:

$$\sup_{p^* \in \mathcal{K}} \left\{ - \left\langle \frac{-p_2^*}{4\alpha_2} + g_2, -p_2^* \right\rangle_{\mathcal{H}} - \langle g_1, -p_1^* \rangle_{\mathcal{H}} - \varphi_+^*(|p_0^*|)(\Omega) \right\}, \quad (2.22)$$

where  $\mathcal{K} = \{p^* \in \mathcal{H}^d \times \mathcal{H} \times \mathcal{H} : |p_0^*(x)| \in \operatorname{Dom} \varphi_+^* \text{ and } |p_1^*(x)| \leq \alpha_1 \forall x \in \Omega, \Lambda^* p^* + \zeta = 0\}$ . The function  $\varphi_+$  also fulfills assumption  $(A_\varphi)$ (ii) (i.e., there exist  $c_1 > 0$ ,  $b \geq 0$  such that  $c_+ z - b \leq \varphi_+(z) \leq c_+ z + b$  for all  $z \in \mathbb{R}^+$ ). The conjugate function of  $\varphi_+$  is given by  $\varphi_+^*(t) = \sup_{z \in \mathbb{R}} \{\langle t, z \rangle - \varphi_+(z)\}$ . Using the previous inequalities and the fact that  $\varphi_+$  is an even function (i.e.,  $\varphi_+(z) = \varphi_+(-z)$  for all  $z \in \mathbb{R}$ ), we have

$$\sup_{z \in \mathbb{R}} \{\langle t, z \rangle - c_+ |z| + b\} \geq \sup_{z \in \mathbb{R}} \{\langle t, z \rangle - \varphi_+(z)\} \geq \sup_{z \in \mathbb{R}} \{\langle t, z \rangle - c_1 |z| - b\} = \begin{cases} -b & \text{if } |t| \leq c_1, \\ \infty & \text{else.} \end{cases} \quad (2.23)$$

In particular, one can see that  $t \in \operatorname{Dom} \varphi_+$  if and only if  $|t| \leq c_1$ .

## I. Subspace Correction Methods for Total Variation Minimization

From  $\Lambda^* p^* + \zeta = 0$  we obtain

$$\begin{aligned} \langle \Lambda^* p^*, \omega \rangle_{\mathcal{H}} + \langle \zeta, \omega \rangle_{\mathcal{H}} &= \langle p^*, \Lambda \omega \rangle_{\mathcal{H}^{d+2}} + \langle \zeta, \omega \rangle_{\mathcal{H}} \\ &= \langle p_0^*, \nabla \omega \rangle_{\mathcal{H}^d} + \langle p_1^*, T_1 \omega \rangle_{\mathcal{H}} + \langle p_2^*, T_2 \omega \rangle_{\mathcal{H}} + \langle \zeta, \omega \rangle_{\mathcal{H}} = 0 \quad \forall \omega \in \mathcal{H}. \end{aligned} \quad (2.24)$$

Then, since  $\langle p_0^*, \nabla \omega \rangle_{\mathcal{H}^d} = \langle -\operatorname{div} p_0^*, \omega \rangle_{\mathcal{H}}$  (see Section 2.4.1), we have  $T_1^* p_1^* + T_2^* p_2^* - \operatorname{div} p_0^* + \zeta = 0$ . Hence we can write  $\mathcal{K}$  in the following way:

$$\mathcal{K} = \left\{ p^* = (p_0^*, p_1^*, p_2^*) \in \mathcal{H}^d \times \mathcal{H} \times \mathcal{H} : |p_0^*(x)| \leq c_1 \text{ and } |p_1^*(x)| \leq \alpha_1 \quad \forall x \in \Omega, \right. \\ \left. T_1^* p_1^* + T_2^* p_2^* - \operatorname{div} p_0^* + \zeta = 0 \right\}.$$

We now apply the duality results from [26, Thm. III.4.1], since the objective functional in  $(\mathcal{P})$  is convex, continuous with respect to  $\Lambda v$  in  $\mathcal{H}^d \times \mathcal{H} \times \mathcal{H}$ , and  $\inf(\mathcal{P})$  is finite. Consequently,  $\inf(\mathcal{P}) = \sup(\mathcal{P}^*) \in \mathbb{R}$ , and  $(\mathcal{P}^*)$  has a solution  $M = (M_0, M_1, M_2) \in \mathcal{K}$ .

Let us assume that  $u$  is a solution of  $(\mathcal{P})$  and  $M$  is a solution of  $(\mathcal{P}^*)$ . From  $\inf(\mathcal{P}) = \sup(\mathcal{P}^*)$  we get

$$\begin{aligned} \alpha_1 \|T_1 u - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{\ell^2(\Omega)}^2 + \varphi(|\nabla u|)(\Omega) - \langle \zeta, u \rangle_{\mathcal{H}} \\ = - \left\langle \frac{-M_2}{4\alpha_2} + g_2, -M_2 \right\rangle_{\mathcal{H}} - \langle g_1, -M_1 \rangle_{\mathcal{H}} - \varphi_1^* (|M_0|)(\Omega), \end{aligned} \quad (2.25)$$

where  $M = (M_0, M_1, M_2) \in \mathcal{H}^d \times \mathcal{H} \times \mathcal{H}$ ,  $|M_0(x)| \leq c_1$ ,  $|M_1| \leq \alpha_1$ , and  $T_1^* M_1 + T_2^* M_2 - \operatorname{div} M_0 + \zeta = 0$ , which verifies (2.20). In particular, (2.24) and (2.25) yield

$$\begin{aligned} \alpha_1 \|T_1 u - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{\ell^2(\Omega)}^2 + \varphi(|\nabla u|)(\Omega) + \langle M_1, T_1 u \rangle_{\mathcal{H}} + \langle M_2, T_2 u \rangle_{\mathcal{H}} \\ + \langle M_0, \nabla u \rangle_{\mathcal{H}^d} + \left\langle \frac{-M_2}{4\alpha_2} + g_2, -M_2 \right\rangle_{\mathcal{H}} + \langle g_1, -M_1 \rangle_{\mathcal{H}} + \varphi_1^* (|M_0|)(\Omega) = 0. \end{aligned} \quad (2.26)$$

We rewrite (2.26) in the following form:

$$\begin{aligned} \sum_{x \in \Omega} \alpha_1 |(T_1 u - g_1)(x)| + \sum_{x \in \Omega} \alpha_2 |(T_2 u - g_2)(x)|^2 + \sum_{x \in \Omega} \varphi(|(\nabla u)(x)|) + \sum_{x \in \Omega} M_1(x)(T_1 u)(x) \\ + \sum_{x \in \Omega} M_2(x)(T_2 u)(x) + \sum_{x \in \Omega} \langle M_0(x), \nabla u(x) \rangle_{\mathbb{R}^d} - \sum_{x \in \Omega} \left( \frac{-M_2(x)}{4\alpha_2} + g_2(x) \right) (M_2)(x) \\ + \sum_{x \in \Omega} g_1(x)(-M_1)(x) + \sum_{x \in \Omega} \varphi_1^* (|M_0(x)|) = 0. \end{aligned} \quad (2.27)$$

Now for the various terms in (2.27) we have the following:

1.  $\alpha_1 |(T_1 u - g_1)(x)| + M_1(x)((T_1 u)(x) - g_1(x)) \geq 0$  since  $|M_1(x)| \leq \alpha_1$ .
2.  $\alpha_2 |(T_2 u - g_2)(x)|^2 + M_2(x)(T_2 u(x) - g_2(x)) + \frac{M_2(x)^2}{4\alpha_2} = (\sqrt{\alpha_2}(T_2 u - g_2)(x) + \frac{M_2(x)}{2\sqrt{\alpha_2}})^2 \geq 0$ .
3.  $\varphi(|(\nabla u)(x)|) + \langle M_0(x), \nabla u(x) \rangle_{\mathbb{R}^d} + \varphi_1^* (|M_0(x)|) \geq \varphi(|(\nabla u)(x)|) - \sum_{j=1}^d |M_0^j(x)||S^j| + \varphi_1^* (|M_0(x)|) \geq 0$  by the definition of  $\varphi_1^*$ , since

$$\varphi_1^* (|M_0(x)|) = \sup_{S=(S^1, \dots, S^d) \in \mathbb{R}^d} \left\{ \sum_{j=1}^d |M_0^j(x)||S^j| - \varphi(|S|) \right\}.$$

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

Hence, condition (2.27) reduces to

$$\varphi(|(\nabla u)(x)|) + \langle M_0(x), \nabla u(x) \rangle_{\mathbb{R}^d} + \varphi_1^*(|M_0(x)|) = 0 \quad \forall x \in \Omega, \quad (2.28)$$

$$M_2(x) = -2\alpha_2(T_2 u - g_2)(x) \quad \forall x \in \Omega, \quad (2.29)$$

$$\alpha_1|(T_1 u - g_1)(x)| + M_1(x)((T_1 u)(x) - g_1(x)) = 0 \quad \forall x \in \Omega. \quad (2.30)$$

Conversely, if there exists  $M = (M_0, M_1, M_2) \in \mathcal{H}^d \times \mathcal{H} \times \mathcal{H}$  with  $|M_0(x)| \leq c_1$  and  $|M_1| \leq \alpha_1$ , which fulfills conditions (2.17) and (2.20), then it is clear from our previous considerations that (2.25) holds. Let us denote the functional on the left-hand side of (2.25) by

$$P(u) := \alpha_1 \|T_1 u - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{\ell^2(\Omega)}^2 + \varphi(|\nabla u|)(\Omega) - \langle \zeta, u \rangle_{\mathcal{H}}$$

and the functional on the right-hand side of (2.25) by

$$P^*(M) := - \left\langle \frac{-M_2}{4\alpha_2} + g_2, -M_2 \right\rangle_{\mathcal{H}} - \langle g_1, -M_1 \rangle_{\mathcal{H}} - \varphi_1^*(|M_0|)(\Omega).$$

Hence  $\inf P = \inf(\mathcal{P})$  and  $\sup P^* = \sup(\mathcal{P}^*)$ . Since  $P$  is convex, continuous with respect to  $\Lambda u$  in  $\mathcal{H}^d \times \mathcal{H} \times \mathcal{H}$ , and  $\inf(\mathcal{P})$  is finite, we know from duality results [26, Thm. III.4.1] that  $\inf(\mathcal{P}) = \sup(\mathcal{P}^*) \in \mathbb{R}$ . We assume that  $M$  is no solution of  $(\mathcal{P}^*)$ , i.e.,  $P^*(M) < \sup(\mathcal{P}^*)$ , and  $u$  is no solution of  $(\mathcal{P})$ , i.e.,  $P(u) > \inf(\mathcal{P})$ . Then we have that  $P(u) > \inf(\mathcal{P}) = \sup(\mathcal{P}^*) > P^*(M)$ . Thus (2.25) is valid if and only if  $M$  is a solution of  $(\mathcal{P}^*)$  and  $u$  is a solution of  $(\mathcal{P})$ , which is equivalent to  $\zeta \in \partial J_{\alpha_1, \alpha_2}(u)$ .

If, additionally,  $\varphi$  is differentiable and  $|(\nabla u)(x)| \neq 0$  for  $x \in \Omega$ , then  $M_0(x)$  can be computed explicitly. In fact, from (2.28) (respectively, (2.17)) we have

$$\varphi_1^*(|-M_0(x)|) = -\langle M_0(x), (\nabla u)(x) \rangle_{\mathbb{R}^d} - \varphi(|(\nabla u)(x)|). \quad (2.31)$$

From the definition of conjugate functions we have

$$\begin{aligned} \varphi_1^*(|-M_0(x)|) &= \sup_{t \in \mathbb{R}} \{|-M_0(x)|t - \varphi_1(t)\} = \sup_{t \geq 0} \{|-M_0(x)|t - \varphi_1(t)\} \\ &= \sup_{t \geq 0} \sup_{\substack{S \in \mathbb{R}^d \\ |S|=t}} \{\langle -M_0(x), S \rangle_{\mathbb{R}^d} - \varphi_1(|S|)\} = \sup_{S \in \mathbb{R}^d} \{\langle -M_0(x), S \rangle_{\mathbb{R}^d} - \varphi(|S|)\}. \end{aligned}$$

Now, if  $|(\nabla u)(x)| \neq 0$  for  $x \in \Omega$ , then it follows from (2.31) that the supremum is taken in  $S = |(\nabla u)(x)|$  and we have  $\nabla_S(-\langle M_0(x), S \rangle_{\mathbb{R}^d} - \varphi(|S|)) = 0$ , which implies

$$M_0^j(x) = -\frac{\varphi'(|(\nabla u)(x)|)}{|(\nabla u)(x)|} (\nabla u)^j(x), \quad j = 1, \dots, d,$$

and verifies (2.21). This finishes the proof.  $\square$

For  $\alpha_2 = 0$  the minimization problem associated with the objective in (2.13) becomes

$$\min_{u \in \mathcal{H}} J_{\alpha_1, 0}(u) = \alpha_1 \|(T_1 u - g_1)\|_{\ell^1(\Omega)} + \varphi(|\nabla u|)(\Omega), \quad (2.32)$$

and system (2.17)-(2.20) reduces to (2.33)-(2.35) below.

## I. Subspace Correction Methods for Total Variation Minimization

**Corollary 2.10.** *Let  $\zeta, u \in \mathcal{H}$ . If assumption  $(A_\varphi)$  holds true, then  $\zeta \in \partial J_{\alpha_1,0}(u)$  if and only if there exist  $M = (M_0, M_1) \in \mathcal{H}^d \times \mathcal{H}$  and a constant  $c_1 \geq 0$  with  $|M_0(x)| \leq c_1$ ,  $|M_1(x)| \leq \alpha_1$  for all  $x \in \Omega$  such that*

$$\varphi(|(\nabla u)(x)|) + \langle M_0(x), \nabla u(x) \rangle_{\mathbb{R}^d} + \varphi_1^*(|M_0(x)|) = 0 \quad \forall x \in \Omega, \quad (2.33)$$

$$\alpha_1|(T_1 u - g_1)(x)| + M_1(x)((T_1 u)(x) - g_1(x)) = 0 \quad \forall x \in \Omega, \quad (2.34)$$

$$T_1^* M_1 - \operatorname{div} M_0 + \zeta = 0, \quad (2.35)$$

where  $\varphi_1^*$  is the conjugate function of  $\varphi_1$  defined by  $\varphi_1(t) = \varphi(|t|)$  for  $t \in \mathbb{R}$ .

**Optimality Properties** By [35, Thm. 2.1.4, p. 305], the optimality condition for the subspace minimization problem in  $V_i$  (cf. (2.11)), i.e.,

$$\xi_i^{(n+1)} \in \arg \min_{\xi_i \in \mathcal{H}} \{J_{\alpha_1, \alpha_2}(\xi_i) : \pi_{V_i^c} \xi_i = \pi_{V_i^c} b\}, \quad (2.36)$$

is

$$0 \in \partial J_{\alpha_1, \alpha_2}(\xi_i^{(n+1)}) + \eta_i^{(n+1)},$$

where  $\eta_i^{(n+1)} \in \operatorname{Range}(\pi_{V_i^c}^*) \simeq V_i^c$  and  $b \in V_j$  as in (2.11) for  $i = 1, 2$  and  $j \in \{1, 2\} \setminus \{i\}$ . Note that indeed  $\xi_1^{(n+1)}$  is optimal for (2.36) with  $i = 1$  and  $b = \tilde{u}_2^{(n)}$  if and only if  $u_1^{(n+1)} = \xi_1^{(n+1)} - \tilde{u}_2^{(n)}$  is optimal for the first minimization problem in (2.9) or (2.10), respectively. Moreover,  $\xi_2^{(n+1)}$  is optimal for (2.36) with  $i = 2$  and  $b = u_1^{(n+1)}$  if and only if  $u_2^{(n+1)} = \xi_2^{(n+1)} - u_1^{(n+1)}$  is a solution of the second minimization problem in (2.9), and  $\xi_2^{(n+1)}$  is optimal for (2.36) with  $i = 2$  and  $b = \tilde{u}_1^{(n)}$  if and only if  $u_2^{(n+1)} = \xi_2^{(n+1)} - \tilde{u}_1^{(n)}$  is optimal for the second minimization problem in (2.10).

The following result is a consequence of Proposition 2.9 and Corollary 2.10. It relies on the fact that  $\partial J_{\alpha_1, \alpha_2}(\xi)$  is compact for any  $\xi \in \mathcal{H}$ ; see [6].

**Corollary 2.11.** *Let  $(\xi^{(n)})_n \subset \mathcal{H}$  be bounded and  $\eta^{(n)} \in \partial J_{\alpha_1, \alpha_2}(\xi^{(n)})$  for all  $n \in \mathbb{N}$ . Then  $(\eta^{(n)})_n$  is bounded.*

*Proof.* Set  $P(\xi^{(n)}) := \alpha_1 \|T_1 \xi^{(n)} - g_1\|_{\ell^1(\Omega)} + \varphi(|\nabla \xi^{(n)}|)(\Omega)$ . Then we have

$$\eta^{(n)} \in 2\alpha_2 T_2^*(T_2 \xi^{(n)} - g_2) + \partial P(\xi^{(n)}).$$

Since  $T_2$  is a bounded operator and  $(\xi^{(n)})_n$  is bounded, we are left with showing that the set  $\partial P(\xi^{(n)})$  is bounded for all  $n$ . By Corollary 2.10 we have that

$$\begin{aligned} \partial P(\xi^{(n)}) &= \{ \operatorname{div} M_0 - T_1^* M_1 \in \mathcal{H} : \|M_0\|_\infty \leq c_1, \|M_1\|_\infty \leq \alpha_1, \\ &\quad \varphi(|(\nabla \xi^{(n)})(x)|) + \langle M_0(x), \nabla \xi^{(n)}(x) \rangle_{\mathbb{R}^d} + \varphi_1^*(|M_0(x)|) = 0, \\ &\quad \alpha_1 |(T_1 \xi^{(n)} - g_1)(x)| + M_1(x)((T_1 \xi^{(n)})(x) - g_1(x)) = 0 \text{ for all } x \in \Omega \}. \end{aligned}$$

Since  $c_1$  and  $\alpha_1$  do not depend on  $n$ , the sequence of sets  $(\partial P(\xi^{(n)}))_n$  is uniformly bounded, and hence  $(\eta^{(n)})_n$  is bounded as well.  $\square$

**Proposition 2.12.** *There exist accumulation points  $u_i^{(\infty)}$  and  $\tilde{u}_i^{(\infty)}$  of the sequences  $(u_i^{(n)})_n$  and  $(\tilde{u}_i^{(n)})_n$ ,  $i = 1, 2$ , generated by the algorithms in (2.9) or (2.10) such that*

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

- (i)  $u_1^{(\infty)}$  and  $\tilde{u}_1^{(\infty)}$  are minimizers of  $\min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(\infty)})$  and
- (ii) for the algorithm in (2.9),  $u_2^{(\infty)}$  and  $\tilde{u}_2^{(\infty)}$  are minimizers of  $\min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(u_1^{(\infty)} + u_2)$ , and for the algorithm in (2.10),  $u_2^{(\infty)}$  and  $\tilde{u}_2^{(\infty)}$  are minimizers of  $\min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(\infty)} + u_2)$ .

*Proof.* We start by showing the assertion for algorithm (2.9). By Proposition 2.4 the sequence  $(\tilde{u}_2^{(n)})_n$  is bounded and hence has a convergent subsequence  $(\tilde{u}_2^{(n_k)})_k$  with limit  $\tilde{u}_2^{(\infty)}$ . Further,  $(u_1^{(n_k+1)})_k$  is bounded and has a subsequence  $(u_1^{(n_{k_\ell}+1)})_\ell$  which converges to  $u_1^{(\infty)}$ . The minimality property of  $u_1^{(n_{k_\ell}+1)}$  yields that  $0 \in \partial_{V_1} J_{\alpha_1, \alpha_2}(\cdot + \tilde{u}_2^{(n_{k_\ell})})(u_1^{(n_{k_\ell}+1)})$ . By [44, Thm. 24.4, p 233] we obtain that  $0 \in \partial_{V_1} J_{\alpha_1, \alpha_2}(\cdot + \tilde{u}_2^{(\infty)})(u_1^{(\infty)})$  and hence  $u_1^{(\infty)} \in \arg \min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(\infty)})$ . Since  $(u_2^{(n_{k_\ell}+1)})_\ell$  is again bounded, there exists a convergent subsequence  $(u_2^{(n_{k_{\ell_j}}+1)})_j$  with limit  $u_2^{(\infty)}$ , and we get that  $0 \in \partial_{V_2} J_{\alpha_1, \alpha_2}(u_1^{(\infty)} + \cdot)(u_2^{(\infty)})$ , i.e.,  $u_2^{(\infty)} = \arg \min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(u_1^{(\infty)} + u_2)$ . Moreover, by the monotone decrease of the energy (see Proposition 2.2), we have that

$$J_{\alpha_1, \alpha_2}(u^{(n)}) \geq J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + \tilde{u}_2^{(n)}) \geq J_{\alpha_1, \alpha_2}(u^{(n+1)}) \quad \forall n \in \mathbb{N}, \quad (2.37)$$

and hence

$$J_{\alpha_1, \alpha_2}(u^{(n)}) - J_{\alpha_1, \alpha_2}(u^{(n+1)}) \geq J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + \tilde{u}_2^{(n)}) - J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(n+1)} + \tilde{u}_2^{(n+1)}) \geq 0$$

as well as

$$J_{\alpha_1, \alpha_2}(u^{(n)}) - J_{\alpha_1, \alpha_2}(u^{(n+1)}) \geq J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + \tilde{u}_2^{(n)}) - J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + u_2^{(n+1)}) \geq 0.$$

Since  $J_{\alpha_1, \alpha_2}$  is bounded from below, we obtain  $\lim_{n \rightarrow \infty} [J_{\alpha_1, \alpha_2}(u^{(n)}) - J_{\alpha_1, \alpha_2}(u^{(n+l)})] = 0$  for any  $l \in \mathbb{N}$ . Consequently

$$0 = \lim_{n_{k_\ell} \rightarrow \infty} \left[ J_{\alpha_1, \alpha_2} \left( u_1^{(n_{k_\ell}+1)} + \tilde{u}_2^{(n_{k_\ell})} \right) - J_{\alpha_1, \alpha_2} \left( \tilde{u}_1^{(n_{k_\ell}+1)} + \tilde{u}_2^{(n_{k_\ell}+1)} \right) \right] \quad (2.38)$$

$$\begin{aligned} 0 &= \lim_{n_{k_{\ell_j}} \rightarrow \infty} \left[ J_{\alpha_1, \alpha_2} \left( u_1^{(n_{k_{\ell_j}}+1)} + \tilde{u}_2^{(n_{k_{\ell_j}})} \right) - J_{\alpha_1, \alpha_2} \left( u_1^{(n_{k_{\ell_j}}+1)} + u_2^{(n_{k_{\ell_j}}+1)} \right) \right] \\ &= J_{\alpha_1, \alpha_2} \left( u_1^{(\infty)} + \tilde{u}_2^{(\infty)} \right) - J_{\alpha_1, \alpha_2} \left( u_1^{(\infty)} + u_2^{(\infty)} \right). \end{aligned} \quad (2.39)$$

From (2.39) we observe that since  $u_2^{(\infty)} \in \arg \min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(u_1^{(\infty)} + u_2)$ , then also  $\tilde{u}_2^{(\infty)} \in \arg \min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(u_1^{(\infty)} + u_2)$ .

Since  $(\tilde{u}_1^{(n_{k_\ell+1})})_\ell$  is bounded, there exists a convergent subsequence  $(\tilde{u}_1^{(n_{k_{\ell_m+1}})})_m$  with limit  $\tilde{u}_1^{(\infty)}$ , and we obtain from (2.38)

$$\begin{aligned} 0 &= \lim_{n_{k_{\ell_m}} \rightarrow \infty} \left[ J_{\alpha_1, \alpha_2} \left( u_1^{(n_{k_{\ell_m}}+1)} + \tilde{u}_2^{(n_{k_{\ell_m}})} \right) - J_{\alpha_1, \alpha_2} \left( \tilde{u}_1^{(n_{k_{\ell_m+1}})} + \tilde{u}_2^{(n_{k_{\ell_m+1}})} \right) \right] \\ &= J_{\alpha_1, \alpha_2}(u_1^{(\infty)} + \tilde{u}_2^{(\infty)}) - J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(\infty)} + \tilde{u}_2^{(\infty)}). \end{aligned} \quad (2.40)$$

## I. Subspace Correction Methods for Total Variation Minimization

From (2.40) we infer that since  $u_1^{(\infty)}$  is a solution of  $\min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(\infty)})$ ,  $\tilde{u}_1^{(\infty)}$  is a solution as well.

Although the proof for the algorithm in (2.10) is similar, for the sake of completeness we provide the details here. By similar arguments as above, one shows that there exist accumulation points  $u_1^{(\infty)} \in \arg \min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(\infty)})$  and  $u_2^{(\infty)} = \arg \min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(\infty)} + u_2)$ . Hence, we have

$$\begin{aligned} J_{\alpha_1, \alpha_2}(u_1^{(\infty)} + \tilde{u}_2^{(\infty)}) &\leq J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(\infty)}) \quad \forall u_1 \in V_1, \\ J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(\infty)} + u_2^{(\infty)}) &\leq J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(\infty)} + u_2) \quad \forall u_2 \in V_2. \end{aligned} \quad (2.41)$$

Moreover, by the monotone decrease of the energy  $J_{\alpha_1, \alpha_2}$  (see Proposition 2.2), we have

$$J_{\alpha_1, \alpha_2}(u^{(n)}) \geq \frac{1}{2} \left( J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + \tilde{u}_2^{(n)}) + J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(n)} + u_2^{(n+1)}) \right) \geq J_{\alpha_1, \alpha_2}(u^{(n+1)})$$

and hence

$$\begin{aligned} 2(J_{\alpha_1, \alpha_2}(u^{(n)}) - J_{\alpha_1, \alpha_2}(u^{(n+1)})) \\ \geq J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + \tilde{u}_2^{(n)}) + J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(n)} + u_2^{(n+1)}) - 2J_{\alpha_1, \alpha_2}(u^{(n+1)}) \geq 0. \end{aligned}$$

Since  $J_{\alpha_1, \alpha_2}$  is bounded below, we obtain  $\lim_{n \rightarrow \infty} [J_{\alpha_1, \alpha_2}(u^{(n)}) - J_{\alpha_1, \alpha_2}(u^{(n+l)})] = 0$  for any  $l \in \mathbb{N}$ , and hence

$$\begin{aligned} \lim_{n \rightarrow \infty} \left[ J_{\alpha_1, \alpha_2} \left( u_1^{(n_{k_{\ell_{j_m}}+1})} + \tilde{u}_2^{(n_{k_{\ell_{j_m}}})} \right) + J_{\alpha_1, \alpha_2} \left( \tilde{u}_1^{(n_{k_{\ell_{j_m}}})} + u_2^{(n_{k_{\ell_{j_m}}+1})} \right) \right. \\ \left. - 2J_{\alpha_1, \alpha_2} \left( \tilde{u}_1^{(n_{k_{\ell_{j_m+1}}})} + \tilde{u}_2^{(n_{k_{\ell_{j_m+1}}})} \right) \right] = 0. \end{aligned}$$

From (2.41) it follows that

$$J_{\alpha_1, \alpha_2}(u_1^{(\infty)} + \tilde{u}_2^{(\infty)}) - J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(\infty)} + \tilde{u}_2^{(\infty)}) = 0 \quad \text{and} \quad J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(\infty)} + u_2^{(\infty)}) - J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(\infty)} + \tilde{u}_2^{(\infty)}) = 0,$$

and hence

$$\tilde{u}_1^{(\infty)} \in \arg \min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(\infty)}) \quad \text{and} \quad \tilde{u}_2^{(\infty)} \in \arg \min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(\tilde{u}_1^{(\infty)} + u_2),$$

which concludes the proof. □

**Remark 2.13.** (i) If  $V_1$  and  $V_2$  are disjoint spaces, then the algorithm in (2.9) generates sequences  $(u_i^{(n)})_n$  and  $(\tilde{u}_i^{(n)})_n$  with  $u_i^{(n)} = \tilde{u}_i^{(n)}$ , and the algorithm in (2.10) generates sequences  $(u_i^{(n)})_n$  and  $(\tilde{u}_i^{(n)})_n$  with  $\tilde{u}_i^{(n+1)} = \frac{1}{2}(u_i^{(n+1)} + u_i^{(n)})$ ,  $\tilde{u}_i^{(\infty)} = u_i^{(\infty)}$ , and  $u^{(\infty)} = u_1^{(\infty)} + u_2^{(\infty)}$ .

(ii) In general, however, the algorithms in (2.9) and (2.10), respectively, may generate sequences  $(u_i^{(n)})_n$  and  $(\tilde{u}_i^{(n)})_n$  with  $u_i^{(n)} \neq \tilde{u}_i^{(n)}$ ,  $i = 1, 2$ . This relation is still valid in the limit case; i.e., for  $n \rightarrow \infty$  we have  $u_i^{(\infty)} \neq \tilde{u}_i^{(\infty)}$  (unless  $V_1$  and  $V_2$  are disjoint), although  $u_i^{(\infty)}$  and  $\tilde{u}_i^{(\infty)}$  are minimizers of the same minimization problem; see Proposition 2.4. This behavior can be attributed to the fact that  $J_{\alpha_1, \alpha_2}$  is not strictly convex and, thus, has in general more than one minimizer.

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

Next we provide an estimate for the distance between an accumulation point generated by one of the algorithms (2.9) and (2.10) and a minimizer of  $J_{\alpha_1, \alpha_2}$ .

**Theorem 2.14.** *Assume that  $u^*$  is a minimizer of  $J_{\alpha_1, \alpha_2}$ , and let  $u^{(\infty)}$  be an accumulation point of the sequence  $(u^{(n)})_n$  generated by the algorithm in (2.9) or (2.10). Then we have that*

1.  $u^{(\infty)}$  is a minimizer of  $J_{\alpha_1, \alpha_2}$ , or
2. there exists a constant  $\beta > 0$  such that  $\|u^{(\infty)} - u^*\|_{\ell^2(\Omega)} \leq \beta$ , or
3. if  $\alpha_2 > 0$  and  $T_2^* T_2$  is positive definite in the direction  $u^{(\infty)} - u^*$  with smallest eigenvalue  $\sigma > 0$ , i.e.,  $\|T_2(u^{(\infty)} - u^*)\|_{\ell^2(\Omega)}^2 \geq \sigma \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}^2$ , then

$$\|u^{(\infty)} - u^*\|_{\ell^2(\Omega)} \leq \frac{\|\hat{\eta}\|_{\ell^2(\Omega)}}{\alpha_2 \sigma}, \quad (2.42)$$

where  $\hat{\eta} \in \arg \min_{\eta \in \bigcup_{i=1}^2 (\partial J_{\alpha_1, \alpha_2}(u^{(\infty)}) \cap V_i^c)} \|\eta\|_{\ell^2(\Omega)}$ .

*Proof.* Let  $\xi_i^{(n)} \in \arg \min_{\xi_i \in \mathcal{H}} \{J_{\alpha_1, \alpha_2}(\xi_i) : \pi_{V_i^c} \xi_i = \pi_{V_i^c} b\}$  with  $b \in V_j$  as in (2.11) for  $i = 1, 2$  and  $j \in \{1, 2\} \setminus \{i\}$ . Then, by [35, Thm. 2.1.4, p. 305] we have that  $0 \in \partial J_{\alpha_1, \alpha_2}(\xi_i^{(n)}) + \eta_i^{(n)}$  for  $\eta_i \in V_i^c$  and  $i = 1, 2$ . In other words, for example, for  $i = 1$  we have that  $\xi_1^{(n)} - \tilde{u}_2^{(n-1)} =: u_1^{(n)} \in \arg \min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(n-1)})$ . Since  $(\xi_i^{(n)})_n$  is bounded,  $(\partial J_{\alpha_1, \alpha_2}(\xi_i^{(n)}))_n$  is bounded as well, and (cf. [44, Thm. 24.4, p 233]) there exists  $\eta_i^{(\infty)} \in V_i^c$  such that  $0 \in \partial J_{\alpha_1, \alpha_2}(\xi_i^{(\infty)}) + \eta_i^{(\infty)}$ , where  $\xi_i^{(\infty)}$  is an accumulation point of  $(\xi_i^{(n)})_n$ . Hence,  $u_i^{(\infty)}$  is optimal in  $V_i$  for  $i = 1, 2$ . From Proposition 2.12 we get that if  $u_i^{(\infty)}$  is optimal, then also  $\tilde{u}_i^{(\infty)}$  is optimal. Hence for  $i = 1$  we have  $\tilde{u}^{(\infty)} - \tilde{u}_2^{(\infty)} := \tilde{u}_1^{(\infty)} \in \arg \min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + \tilde{u}_2^{(\infty)})$ , which means that there exists  $\hat{\eta}_1 \in V_1^c$  such that  $0 \in \partial J_{\alpha_1, \alpha_2}(\tilde{u}^{(\infty)}) + \hat{\eta}_1$ . Similarly we get for  $i = 2$  that there exists  $\hat{\eta}_2 \in V_2^c$  such that  $0 \in \partial J_{\alpha_1, \alpha_2}(u^{(\infty)}) + \hat{\eta}_2$ . Note that for the sequential algorithm in (2.9) we have  $\tilde{u}^{(\infty)} \neq u^{(\infty)}$  in general, while for the parallel algorithm in (2.10) we have  $\tilde{u}^{(\infty)} = u^{(\infty)}$ . In the rest of the proof we denote by  $u^{(\infty)}$  an accumulation point which fulfills at least one of the aforementioned inclusions.

By the definition of the subdifferential we obtain

$$J_{\alpha_1, \alpha_2}(u^{(\infty)}) \leq J_{\alpha_1, \alpha_2}(v) + \langle \hat{\eta}, u^{(\infty)} - v \rangle_{\mathcal{H}} \leq J_{\alpha_1, \alpha_2}(v) + \|\hat{\eta}\|_{\ell^2(\Omega)} \|v - u^{(\infty)}\|_{\ell^2(\Omega)} \quad (2.43)$$

for all  $v \in \mathcal{H}$ , where  $\hat{\eta} = \arg \min_{\eta \in \bigcup_{i=1}^2 (\partial J_{\alpha_1, \alpha_2}(u^{(\infty)}) \cap V_i^c)} \|\eta\|_{\ell^2(\Omega)}$ .

Let  $u^* \in \arg \min_{u \in \mathcal{H}} J_{\alpha_1, \alpha_2}(u)$ . Then the optimality of  $u^*$  yields that the directional derivative of  $J_{\alpha_1, \alpha_2}$  at  $u^*$  in any direction  $s \in \mathcal{H}$  is nonnegative; i.e.,  $J'_{\alpha_1, \alpha_2}(u^*; s) \geq 0$ . Set  $P(\xi) := \alpha_1 \|T_1 \xi - g_1\|_{\ell^1(\Omega)} + \varphi(|\nabla \xi|)(\Omega)$ . Then, by using Taylor's expansion, for  $s \in \mathcal{H}$  we have that

$$\begin{aligned} J_{\alpha_1, \alpha_2}(u^* + s) &= \alpha_2 \|T_2 u^* - g_2\|_{\ell^2(\Omega)}^2 + \langle s, 2\alpha_2 T_2^*(T_2 u^* - g_2) \rangle_{\mathcal{H}} + \alpha_2 \|T_2 s\|_{\ell^2(\Omega)}^2 + P(u^* + s) \\ &= J_{\alpha_1, \alpha_2}(u^*) + \langle s, 2\alpha_2 T_2^*(T_2 u^* - g_2) \rangle_{\mathcal{H}} + P(u^* + s) - P(u^*) + \alpha_2 \|T_2 s\|_{\ell^2(\Omega)}^2. \end{aligned}$$

By using  $P(u^* + s) - P(u^*) \geq P'(u^*; s)$ , which easily follows from the convexity of  $P$ , we obtain that

$$J_{\alpha_1, \alpha_2}(u^* + s) \geq J_{\alpha_1, \alpha_2}(u^*) + \langle s, 2\alpha_2 T_2^*(T_2 u^* - g_2) \rangle_{\mathcal{H}} + P'(u^*; s) + \alpha_2 \|T_2 s\|_{\ell^2(\Omega)}^2. \quad (2.44)$$

Since  $J'_{\alpha_1, \alpha_2}(u^*; s) = \langle s, 2\alpha_2 T_2^*(T_2 u^* - g_2) \rangle_{\mathcal{H}} + P'(u^*; s) \geq 0$  and  $\alpha_2 \|T_2 s\|_{\ell^2(\Omega)}^2 \geq 0$  there exists a constant  $\rho \geq 0$  such that  $J_{\alpha_1, \alpha_2}(u^* + s) = J_{\alpha_1, \alpha_2}(u^*) + \rho$ .

## I. Subspace Correction Methods for Total Variation Minimization

1. If  $\rho = 0$  for  $s := u^{(\infty)} - u^*$ , then it immediately follows that  $u^* + s = u^{(\infty)}$  is a minimizer of  $J_{\alpha_1, \alpha_2}$ .
2. If  $\rho > 0$  for  $s := u^{(\infty)} - u^*$ , then from the coercivity condition we obtain that there exists a constant  $\beta > 0$  such that  $0 < \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)} \leq \beta < +\infty$ .
3. If, additionally,  $\alpha_2 > 0$  and there exists a constant  $\sigma > 0$  such that  $\|T_2(u^{(\infty)} - u^*)\|_{\ell^2(\Omega)}^2 \geq \sigma \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}^2$ , then we get from (2.44) that

$$J_{\alpha_1, \alpha_2}(u^* + s) \geq J_{\alpha_1, \alpha_2}(u^*) + \alpha_2 \sigma \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}^2. \quad (2.45)$$

Setting  $v = u^*$  in (2.43) and using the inequality in (2.45) yields

$$J_{\alpha_1, \alpha_2}(u^*) + \alpha_2 \sigma \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}^2 \leq J_{\alpha_1, \alpha_2}(u^{(\infty)}) \leq J_{\alpha_1, \alpha_2}(u^*) + \|\hat{\eta}\|_{\ell^2(\Omega)} \|u^* - u^{(\infty)}\|_{\ell^2(\Omega)}$$

and consequently

$$\|u^{(\infty)} - u^*\|_{\ell^2(\Omega)} \leq \frac{\|\hat{\eta}\|_{\ell^2(\Omega)}}{\alpha_2 \sigma}.$$

□

We have the following immediate consequence of Theorem 2.14.

**Corollary 2.15.** *Let the assumptions of Theorem 2.14 hold true. If  $\|\eta_i^{(n_\ell)}\| \rightarrow 0$  for  $\ell \rightarrow \infty$  along a suitable subsequence  $(n_\ell)_\ell$  for at least one  $i \in \{1, 2\}$ , then any accumulation point of the sequence  $(u^{(n)})_n$  generated by the algorithm in (2.9) or (2.10) is a minimizer of  $J_{\alpha_1, \alpha_2}$ .*

### 2.4.2.2. A Modified Sequential Subspace Correction Method

Note that the algorithm in (2.9) is a special case of a coordinate descent method, where the spaces  $V_i$  are chosen in a cyclic manner; see [48, 49] for more details on different rules for choosing the subspaces. In the sense of coordinate descent methods, our algorithm in (2.9) can be written as follows.

**CD-Algorithm:** Choose  $u^{(0)} \in \mathcal{H}$  and iterate for  $n = 0, 1, 2, \dots$

- (1) choose a nonempty space  $V_n \subset \mathcal{H}$ ;
- (2) compute  $s^{(n)} = s_{T_2^* T_2}(u^{(n)}, V_n) \in \arg \min_s \{J_{\alpha_1, \alpha_2}(u^{(n)} + s) \text{ s.t. } \pi_{V_n^c} s = 0\}$ ;
- (3) set  $u^{(n+1)} = u^{(n)} + s^{(n)}$ .

In step 2, the subscript  $T_2^* T_2$  refers to the Hessian of the smooth part of  $J_{\alpha_1, \alpha_2}$ . Compared to [49], here we choose the step-size in the update of  $u^{(n+1)}$  to be 1, which is justified by (2.15) for  $\alpha_2 > 0$ . As already mentioned, there exist several different ways of choosing  $V_n$  in each iteration. We suggest selecting  $V_n$  such that

$$\|s_D(u^{(n)}, V_n)\|_{\ell^2(\Omega)} \geq \nu \|s_D(u^{(n)}, \mathcal{H})\|_{\ell^2(\Omega)}, \quad (2.46)$$

where  $0 < \nu \leq 1$  and  $D : \mathcal{H} \rightarrow \mathcal{H}$  is positive definite and diagonal, i.e., there exists a  $\tilde{D} \in \mathcal{H}$  associated with  $D$  such that  $Du = \tilde{D} \circ u$  (Hadamard product) with  $[\tilde{D} \circ u](x) = \tilde{D}(x)u(x)$  for

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

any  $u \in \mathcal{H}$  and  $x \in \Omega$ . Here, the subscript  $D$  indicates that  $T_2^*T_2$ , the Hessian of the smooth part of  $J_{\alpha_1, \alpha_2}$ , is replaced by  $D$ . This rule is called the *Gauss–Southwell-r* rule, which also allows the choice  $V_n = \mathcal{H}$ . With this selection rule for the subspaces, we are able to establish global convergence. The proof follows from Theorem 1 of [49].

**Theorem 2.16.** *Assume  $2\alpha_2\|T_2\|^2 \geq \lambda > 0$ . Let  $(u^{(n)})_n, (s^{(n)})_n$  be sequences generated by the CD-Algorithm. If  $(V_n)$  is chosen by the Gauss–Southwell-r rule with  $D$  positive definite, diagonal, and bounded, i.e., there exists a  $\bar{\delta} > 0$  such that  $\|Du\| \leq \bar{\delta}\|u\|$  for all  $u \in \mathcal{H}$ , then every cluster point of  $(u^{(n)})_n$  is a minimizer of  $J_{\alpha_1, \alpha_2}$ .*

## 2.5. Application: Domain Decomposition

The results of the previous sections are valid for any splitting of the function space  $\mathcal{H}$ . We concentrate now on decompositions which split the spatial domain into two subdomains. But let us emphasize that a generalization to a splitting into more domains is straightforward.

### 2.5.1. Overlapping Domain Decomposition

In this section we focus on an overlapping domain decomposition method. Thus we want to minimize (2.13) by decomposing  $\Omega$  into two overlapping subdomains  $\Omega_1$  and  $\Omega_2$  such that  $\Omega = \Omega_1 \cup \Omega_2$  and  $\Omega_1 \cap \Omega_2 \neq \emptyset$ . By  $\Gamma_1$  we denote the interface between  $\Omega_1$  and  $\Omega_2 \setminus \Omega_1$  and by  $\Gamma_2$  the interface between  $\Omega_2$  and  $\Omega_1 \setminus \Omega_2$ . For consistency with the definitions of the gradient and divergence operators, we assume that the subdomains  $\Omega_i$  as well as  $\Omega$  are discrete  $d$ -orthotopes. We stress that this is by no means a restriction, but it simplifies the presentation. Associated with the splitting of  $\Omega$  we define  $V_i = \{u \in \mathcal{H} : \text{supp}(u) \subset \Omega_i\}$ . One aims now to minimize  $J_{\alpha_1, \alpha_2}$  by the alternating algorithm in (2.9) or the parallel algorithm in (2.10). Note that the respective subspace minimization problems are constrained optimization problems of the type (2.11). In particular, for the alternating algorithm we have in  $V_1$

$$\min_{v \in \mathcal{H}} J_{\alpha_1, \alpha_2}(v) \quad \text{s.t. } \pi_{V_1^c} v = \pi_{V_1^c} \tilde{u}_2^{(n)}, \quad (2.47)$$

and in  $V_2$  we have

$$\min_{v \in \mathcal{H}} J_{\alpha_1, \alpha_2}(v) \quad \text{s.t. } \pi_{V_2^c} v = \pi_{V_2^c} u_1^{(n+1)},$$

while for the parallel algorithm the minimization in  $V_1$  is again (2.47), and in  $V_2$  it changes to

$$\min_{v \in \mathcal{H}} J_{\alpha_1, \alpha_2}(v) \quad \text{s.t. } \pi_{V_2^c} v = \pi_{V_2^c} \tilde{u}_1^{(n)}.$$

### 2.5.2. Nonoverlapping Domain Decomposition

In the nonoverlapping domain decomposition method we want to minimize (2.13) by decomposing  $\Omega$  into two nonoverlapping subdomains  $\Omega_1$  and  $\Omega_2$  such that  $\Omega = \Omega_1 \cup \Omega_2$  and  $\Omega_1 = \Omega \setminus \Omega_2$ . For consistency with the definitions of the gradient and divergence operators, we again assume that the subdomains  $\Omega_i$  as well as  $\Omega$  are discrete  $d$ -orthotopes. Associated with the splitting of  $\Omega$  we define by  $V_i = \{u \in \mathcal{H} : \text{supp}(u) \subset \Omega_i\}$  the function space of the subdomain  $\Omega_i$ . Then we minimize  $J_{\alpha_1, \alpha_2}$  either by the parallel algorithm in (2.10) or by the alternating algorithm in (2.9), which specifies to the following.

## I. Subspace Correction Methods for Total Variation Minimization

Choose an initial  $u^{(0)} =: u_1^{(0)} + u_2^{(0)} \in V_1 \oplus V_2$ , for example  $u^{(0)} = 0$ , and iterate

$$\begin{cases} u_1^{(n+1)} \leftarrow \arg \min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + u_2^{(n)}), \\ u_2^{(n+1)} \leftarrow \arg \min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + u_2), \\ u^{(n+1)} \leftarrow u_1^{(n+1)} + u_2^{(n+1)}. \end{cases} \quad (2.48)$$

The subspace optimization problems for the alternating version are

$$\begin{aligned} \min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + u_2^{(n)}) &= \min_{u_1 \in V_1} \alpha_1 \|T_1 u_1 - (g_1 - T_1 u_2^{(n)})\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 u_1 - (g_2 - T_2 u_2^{(n)})\|_{\ell^2(\Omega)}^2 \\ &\quad + \varphi(|\nabla(u_1 + u_2^{(n)})|)(\Omega) \end{aligned}$$

in  $V_1$  and

$$\begin{aligned} \min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + u_2) &= \min_{u_2 \in V_2} \alpha_1 \|T_1 u_2 - (g_1 - T_1 u_1^{(n+1)})\|_{\ell^1(\Omega)} \\ &\quad + \alpha_2 \|T_2 u_2 - (g_2 - T_2 u_1^{(n+1)})\|_{\ell^2(\Omega)}^2 + \varphi(|\nabla(u_1^{(n+1)} + u_2)|)(\Omega) \end{aligned}$$

in  $V_2$ . Upon adjusting notation, for the parallel algorithm the subspace minimization problems look similar.

A very special situation occurs when  $(T_1 u_2)(x) = 0$  for all  $x \in \Omega$  and  $(T_2 u_1)(x) = 0$  for all  $x \in \Omega$ , which is the case when  $T_i = 1_{\Omega_i} \tilde{T}_i$  with  $\tilde{T}_i : \mathcal{H} \rightarrow \mathcal{H}$  such that for all  $v_j \in V_j$  we have  $\tilde{T}_i v_j \in V_j$  for  $j = 1, 2$  and  $i = 1, 2$  (e.g.,  $\tilde{T}_i = I$  or  $\tilde{T}_i = 1_{\Omega \setminus K}$  with  $K \subset \Omega$ ). Then the above subspace minimization problems simplify to

$$\min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + u_2^{(n)}) = \min_{u_1 \in V_1} \alpha_1 \|T_1 u_1 - g_1\|_{\ell^1(\Omega)} + \varphi(|\nabla(u_1 + u_2^{(n)})|)(\Omega) \quad (2.49)$$

in  $V_1$  and

$$\min_{u_2 \in V_2} J_{\alpha_1, \alpha_2}(u_1^{(n+1)} + u_2) = \min_{u_2 \in V_2} \alpha_2 \|T_2 u_2 - g_2\|_{\ell^2(\Omega)}^2 + \varphi(|\nabla(u_1^{(n+1)} + u_2)|)(\Omega) \quad (2.50)$$

in  $V_2$ .

### 2.5.3. Numerical Implementation

In this section we propose an implementation of the domain decomposition algorithms in (2.9) and (2.10) when the domain is split into overlapping and nonoverlapping subdomains for the particular case  $\varphi(|\nabla u|)(\Omega) = |\nabla u|(\Omega)$  (total variation of  $u$  in  $\Omega$ ), i.e., for the minimization problem

$$\arg \min_{u \in \mathcal{H}} \alpha_1 \|T_1 u - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{\ell^2(\Omega)}^2 + |\nabla u|(\Omega). \quad (2.51)$$

As our implementation works for both the nonoverlapping and overlapping domain decomposition algorithm, we use the notation for the overlapping splitting only. More precisely, in the nonoverlapping case  $\Omega_i \setminus \Omega_i$  turns out to be all of  $\Omega_i$  for  $i = 1, 2$  and  $i \in \{1, 2\} \setminus \{i\}$ , and for a nonoverlapping decomposition we have that  $\Gamma_1 = \Gamma_2$  is the interface between  $\Omega_1$  and  $\Omega_2$ .

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

### 2.5.3.1. Implementation of the Domain Decomposition Algorithms

If we compute the minimizer of the functional (2.13) either via the sequential or parallel nonoverlapping domain decomposition algorithm or via the sequential or parallel overlapping domain decomposition algorithm, then, on each subdomain, we have to solve a problem of the type

$$\min_{\xi \in \mathcal{H}} J_{\alpha_1, \alpha_2}(\xi) \quad \text{s.t. } A\xi = b, \quad (2.52)$$

where  $A : \mathcal{H} \rightarrow \mathcal{H}$  is a linear operator or more precisely an orthogonal projection, i.e.,  $A = \pi_{V_i^c}$  for  $i = 1, 2$ . There exist several numerical methods which efficiently solve (2.52). Instances are the *augmented Lagrangian method* [5, 36] or its variations known as Bregman iterations [43, 55, 56] because of their relation to the Bregman distance [8].

Note that the functional  $J_{\alpha_1, \alpha_2}$  is defined on all of  $\Omega$ . We describe now how one may reduce the dimensionality of the subproblems and solve the resulting problems.

**Subspace Minimization** We consider the minimization problem in  $\Omega_1$  written as

$$\min_{u_1 \in V_1} J_{\alpha_1, \alpha_2}(u_1 + u_2) = \alpha_1 \|T_1(u_1 + u_2) - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2(u_1 + u_2) - g_2\|_{\ell^2(\Omega)}^2 + |\nabla(u_1 + u_2)|(\Omega), \quad (2.53)$$

where  $u_2 \in V_2$  is fixed. In order to compute a minimizer of (2.53) we use the discrete analogue of the algorithm described in Section 2.2.2 in  $V_1$ : (i) we introduce a new variable  $v = T_1(u_1 + u_2) - g_1$ , (ii) we regularize the functional in (2.53) as in (2.3), (iii) analogously to (2.4), we solve

$$\min_{v \in \mathcal{H}} \alpha_1 \|v\|_{\ell^1(\Omega)} + \frac{1}{2\gamma} \|T_1 u - g_1 - v\|_{\ell^2(\Omega)}^2, \quad (2.54)$$

and (iv) instead of (2.6) we minimize

$$\min_{u_1 \in V_1} \frac{1}{2\gamma} \|T_1 u_1 - (g_1 - T_1 u_2) - v\|_{\ell^2(\Omega)}^2 + \alpha_2 \|T_2 u_1 - (g_2 - T_2 u_2)\|_{\ell^2(\Omega)}^2 + |\nabla(u_1 + u_2)|(\Omega). \quad (2.55)$$

Similarly to (2.8), an approximate solution to (2.55) may be computed by the following iterative algorithm: Initialize  $u_1^{(0)} \in V_1$  and iterate

$$u_1^{(\ell+1)} \in \arg \min_{u_1 \in V_1} S(u_1 + u_2, u_1^{(\ell)} + u_2) \quad \text{for } \ell \geq 0. \quad (2.56)$$

Thanks to the splitting property of the total variation, i.e.,

$$|\nabla(u_1 + u_2)|(\Omega) = |\nabla(u_1 + u_2)|(\Omega_1 \cup \tilde{\Omega}_2) + f(u_2), \quad (2.57)$$

where  $f$  is a function independent of  $u_1$  (see [2]), we can restrict (2.56) to  $\Omega_1 \cup \tilde{\Omega}_2$ , where  $\tilde{\Omega}_2 \subset \Omega_2 \setminus \Omega_1$  is a small neighborhood strip around the interface  $\Gamma_1$ . Hence the minimization problem in (2.56) is equivalent to

$$\min_{u_1 \in V_1} \left\| u_1 + u_2 - \frac{\gamma}{1 + 2\alpha_2\gamma} \left( \frac{1}{\gamma} z_1 + 2\alpha_2 z_2 \right) \right\|_{\ell^2(\Omega_1 \cup \tilde{\Omega}_2)}^2 + \frac{2\gamma}{1 + 2\alpha_2\gamma} |\nabla(u_1 + u_2)|(\Omega_1 \cup \tilde{\Omega}_2)$$

## I. Subspace Correction Methods for Total Variation Minimization

with  $z_1 = u_1^{(\ell)} + u_2 + T_1^*(g_1 + v - T_1 u_1^{(\ell)} - T_1 u_2)$  and  $z_2 = u_1^{(\ell)} + u_2 + T_2^*(g_2 - T_2 u_1^{(\ell)} - T_2 u_2)$ . We compute a solution of this problem by solving the constrained minimization problem

$$\begin{aligned} \min_{\xi_1 \in V_1 \oplus \tilde{V}_2} & \left\| \xi_1 - \frac{\gamma}{1+2\alpha_2\gamma} \left( \frac{1}{\gamma} z_1 + 2\alpha_2 z_2 \right) \right\|_{\ell^2(\Omega_1 \cup \tilde{\Omega}_2)}^2 + \frac{2\gamma}{1+2\alpha_2\gamma} |\nabla \xi_1|(\Omega_1 \cup \tilde{\Omega}_2), \\ \text{s.t. } & \pi_{\tilde{V}_2} \xi_1 = \pi_{\tilde{V}_2} u_2, \end{aligned} \quad (2.58)$$

where  $\tilde{V}_2 := \{v \in \mathcal{H} : \text{supp}(v) \subset \tilde{\Omega}_2\}$ . Note that  $\xi_1$  is optimal if and only if  $u_1 = \xi_1 - \pi_{\tilde{V}_2} u_2 \in V_1$  is optimal. We compute a minimizer of the problem in (2.58), which is of the form (2.52), by the Bregmanized operator splitting – split Bregman algorithm [38].

**Remark 2.17.** *The minimization problem (2.4) and (2.54) can be solved very efficiently on the whole domain  $\Omega$ , since we only have to perform a soft-thresholding. On the other hand we could restrict the constrained  $L^2$ -TV minimization (2.58) to the domain  $\Omega_1 \cup \tilde{\Omega}_2$ , i.e., on  $\Omega_1$  plus a small stripe around the interface. This is possible since we freed  $u_1$  from the operators  $T_1$  and  $T_2$  and because of the splitting property of the total variation (2.57).*

**Remark 2.18** ( $L^1$ -TV minimization). *In the case when  $\alpha_2 = 0$  and  $\alpha_1 > 0$ , i.e., the minimization problem in (2.51) becomes the  $L^1$ -TV model, each subspace minimization problem can be computed in the same way as described above. In fact, we first minimize (2.54) and then solve the constrained minimization problem (2.58), which simplifies to*

$$\min_{\xi_1 \in V_1 \oplus \tilde{V}_2} \|\xi_1 - z_1\|_{\ell^2(\Omega_1 \cup \tilde{\Omega}_2)}^2 + 2\gamma |\nabla \xi_1|(\Omega_1 \cup \tilde{\Omega}_2) \quad \text{s.t. } \pi_{\tilde{V}_2} \xi_1 = \pi_{\tilde{V}_2} u_2.$$

**Remark 2.19** (denoising). *If  $T_1 = T_2 = I$ , then we do not need surrogate functionals, and hence we do not have to perform the iterative algorithm (2.56). Instead we restrict (2.55) directly to  $\Omega_1 \cup \tilde{\Omega}_2$  and solve the following constrained minimization problem:*

$$\begin{aligned} \min_{\xi_1 \in V_1 \oplus \tilde{V}_2} & \left\| \xi_1 - \frac{\gamma}{1+2\alpha_2\gamma} \left( \frac{1}{\gamma} (g_1 + v) + 2\alpha_2 g_2 \right) \right\|_{\ell^2(\Omega_1 \cup \tilde{\Omega}_2)}^2 + \frac{2\gamma}{1+2\alpha_2\gamma} |\nabla \xi_1|(\Omega_1 \cup \tilde{\Omega}_2) \\ \text{s.t. } & \pi_{\tilde{V}_2} \xi_1 = \pi_{\tilde{V}_2} u_2. \end{aligned} \quad (2.59)$$

The minimization problem in  $\Omega_2$  can be solved in the same way by adjusting the notations accordingly.

### 2.5.3.2. A Special Case

The implementation of the special case  $T_i = 1_{\Omega_i} \tilde{T}_i$  and  $g_i = 1_{\Omega_i} \tilde{g}_i$ , where  $\tilde{T}_i : \mathcal{H} \rightarrow \mathcal{H}$  and  $\tilde{g}_i \in \mathcal{H}$ , for  $i = 1, 2$ , is considered next. Note that the case considered here is more general than the situation discussed in Section 2.5.2. The minimization problem in (2.51) can be written as

$$\min_{u \in \mathcal{H}} \alpha_1 \|\tilde{T}_1 u - \tilde{g}_1\|_{\ell^1(\Omega_1)} + \alpha_2 \|\tilde{T}_2 u - \tilde{g}_2\|_{\ell^2(\Omega_2)}^2 + |\nabla u|(\Omega).$$

When we solve this problem via one of the suggested domain decomposition methods, then on each subdomain we have to compute the minimizer of a constrained optimization problem. For example, in  $\Omega_1$  we have

$$\min_{u_1 \in V_1} \alpha_1 \|\tilde{T}_1(u_1 + u_2) - \tilde{g}_1\|_{\ell^1(\Omega_1)} + \alpha_2 \|\tilde{T}_2(u_1 + u_2) - \tilde{g}_2\|_{\ell^2(\Omega_2)}^2 + |\nabla(u_1 + u_2)|(\Omega). \quad (2.60)$$

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

A solution of this problem can be obtained as described above in Section 2.5.3.1. However, there exists a more efficient way of solving the problem in  $\Omega_1$ . This strategy, however, is not applicable for the minimization in  $\Omega_2$  due to the special structure of the minimization problems with the  $\ell^1$ -norm defined only on  $\Omega_1$ . The main idea of this more efficient approach is the following one:

1. Free  $u_1$  from the influence of  $\tilde{T}_2$  (respectively  $T_2$ ) by introducing a surrogate functional in a similar way as before; i.e., for  $a \in V_1$  define

$$\begin{aligned} S(u_1 + u_2, a) &:= \alpha_1 \|\tilde{T}_1(u_1 + u_2) - \tilde{g}_1\|_{\ell^1(\Omega_1)} + \alpha_2 \|T_2(u_1 + u_2) - g_2\|_{\ell^2(\Omega)}^2 \\ &\quad + |\nabla(u_1 + u_2)|(\Omega) + \alpha_2 \left( \|u_1 - a\|_{\ell^2(\Omega)}^2 - \|T_2(u_1 - a)\|_{\ell^2(\Omega)}^2 \right) \\ &= \alpha_1 \|\tilde{T}_1(u_1 + u_2) - \tilde{g}_1\|_{\ell^1(\Omega_1)} + \alpha_2 \|u_1 - z_2\|_{\ell^2(\Omega)}^2 + |\nabla(u_1 + u_2)|(\Omega) + \psi, \end{aligned}$$

where  $z_2 = a + T_2^*(g_2 - T_2 u_2 - T_2 a)$  and  $\psi$  is a function independent of  $u_1$ . Then compute an approximate solution of (2.60) by the following algorithm: Initialize  $u_1^{(0)} \in V_1$  and iterate

$$u_1^{(\ell+1)} = \arg \min_{u_1 \in V_1} S(u_1 + u_2, u_1^{(\ell)}) \quad \text{for } \ell \geq 0. \quad (2.61)$$

2. Thanks to (2.57), we can restrict the surrogate functional iteration to  $\Omega_1 \cup \tilde{\Omega}_2$ .
3. In each surrogate iteration (2.61) one has to solve a constrained minimization problem. As an example, we describe here the Bregmanized operator splitting of [56]. For this purpose, let  $\mu, \delta > 0$ , and initialize  $\xi_1^{(0)} \in V_1 \oplus \tilde{V}_2$  and  $b^{(0)} = b = u_2$ . Then for  $k = 0, 1, \dots$  solve

$$\begin{aligned} \xi_1^{(k+1)} &= \arg \min_{\xi_1 \in V_1 \oplus \tilde{V}_2} S(\xi_1, u_1^{(\ell)}) + \frac{\mu}{\delta} \|\xi_1 - (\xi_1^{(k)} - \delta \pi_{\tilde{V}_2}^*(\pi_{\tilde{V}_2} \xi_1^{(k)} - b^{(k)}))\|_{\ell^2(\Omega_1 \cup \tilde{\Omega}_2)}^2, \\ b^{(k+1)} &= b^{(k)} - \pi_{\tilde{V}_2} \xi_1^{(k+1)} + b. \end{aligned} \quad (2.62)$$

4. Solve the minimization problem in (2.62) by the algorithm introduced in Section 2.2.2.

**Remark 2.20.** Practically it seems that recomputing the Bregman update outside of the algorithm of Section 2.2.2 is preferable (to computing the update inside the algorithm of Section 2.2.2, as is done in Section 2.5.3.1), as the resulting overall algorithm seems to converge faster according to our numerical practice.

**Remark 2.21.** In the case when  $\tilde{T}_1 = \tilde{T}_2 = I$  and  $\Omega_1 \cap \Omega_2 = \emptyset$ , on each domain we have to solve the following constrained minimization problems:

$$\min_{u_1 \in V_1 \oplus \tilde{V}_2} \alpha_1 \|u_1 - \tilde{g}_1\|_{\ell^1(\Omega_1 \cup \tilde{\Omega}_2)} + |\nabla(u_1 + u_2)|(\Omega_1 \cup \tilde{\Omega}_2) \quad \text{s.t. } \pi_{\tilde{V}_2} u_1 = 0 \quad (2.63)$$

and

$$\min_{u_2 \in V_2 \oplus \tilde{V}_1} \alpha_2 \|u_2 - \tilde{g}_2\|_{\ell^2(\Omega_2 \cup \tilde{\Omega}_1)}^2 + |\nabla(u_1 + u_2)|(\Omega_2 \cup \tilde{\Omega}_1) \quad \text{s.t. } \pi_{\tilde{V}_1} u_2 = 0, \quad (2.64)$$

where  $u_1 \in V_1$  is fixed in (2.64) and  $\tilde{\Omega}_1$  is defined analogously to  $\tilde{\Omega}_2$ . The subspace minimization problem (2.64) can be solved, for example, by the Bregmanized operator splitting – split Bregman algorithm [38], while one may solve (2.63) as suggested in this section above starting at point 3.

## I. Subspace Correction Methods for Total Variation Minimization

Due to the structure of the problem, for the minimization in  $\Omega_2$  this approach is not applicable, and hence we suggest using the strategy of Section 2.5.3.1.

### 2.5.4. Numerical Experiments

In the following we present numerical experiments for the proposed sequential and parallel algorithms. In particular, we show applications in image denoising, inpainting, and deblurring. The values of the parameters  $\alpha_1$  and  $\alpha_2$  in the objective functional (2.51) are chosen according to the application and experimentally; i.e., we choose the values which give a good compromise between visual quality and computational time of the algorithm. We emphasize that the optimal selection of  $\alpha_1$  and  $\alpha_2$  is an interesting research topic in its own right but goes beyond the scope of the present paper. However, it has been shown in several examples – see [9, 41, 42] – that if only salt-and-pepper noise is present in an image, then the  $L^1$ -TV model outperforms the  $L^2$ -TV model. Hence we use the pure  $L^1$ -TV model when only salt-and-pepper noise has corrupted the image of interest, whereas we use the pure  $L^2$ -TV model when only Gaussian noise is present.

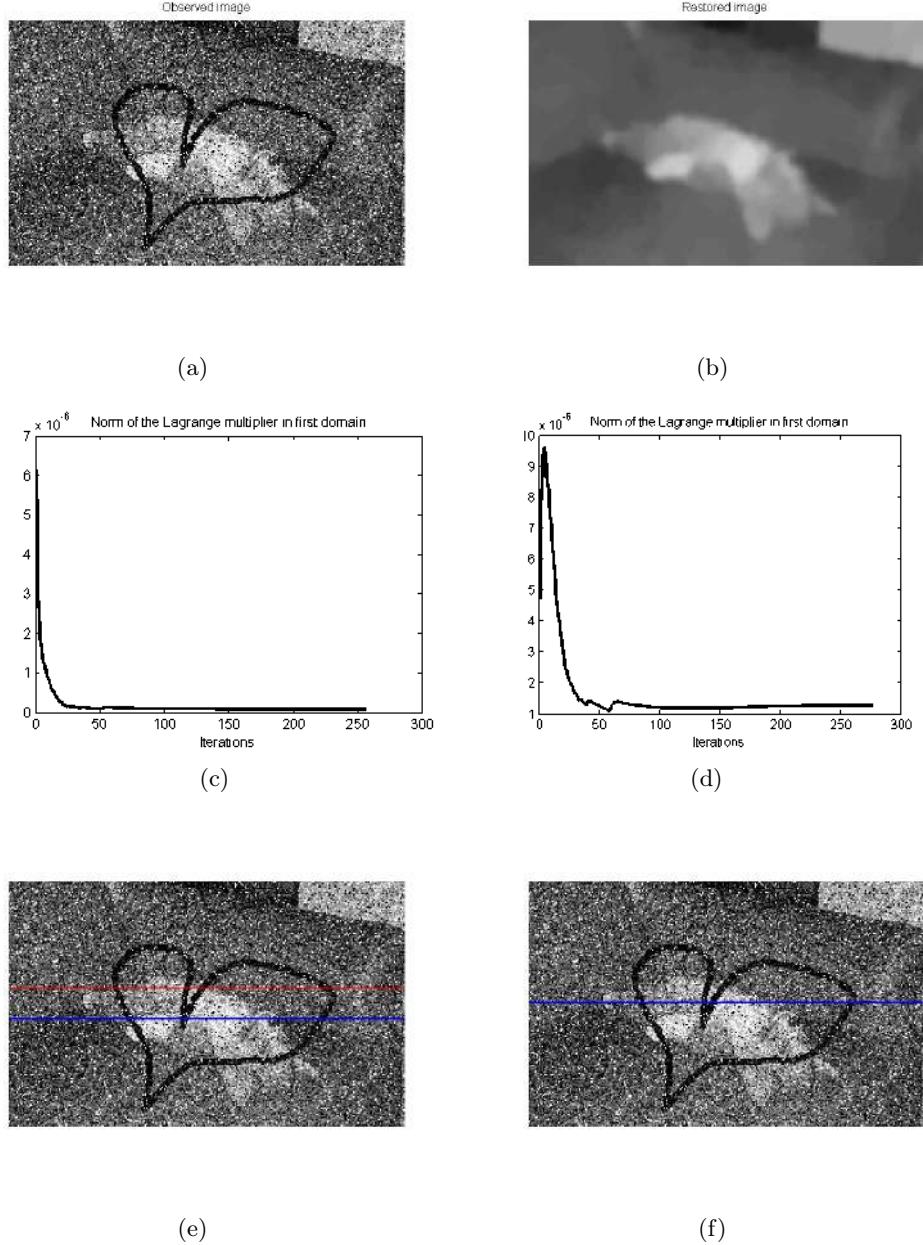
#### 2.5.4.1. Numerical Results – Sequential Algorithms

In our numerical experiments we terminate our sequential algorithms (2.9) and (2.48) as soon as the norm of the difference of two successive iterates drops below a certain threshold. More precisely, we use as a stopping criterion  $\|u^{(n)} - u^{(n+1)}\| < 10^{-6}$ , which seems suitable for our purposes. In fact, if our algorithm converges at least linearly, i.e., if there exists an  $\varepsilon \in (0, 1)$  and an  $m > 0$  such that for all  $n \geq m$  we have  $\|u^{(n+1)} - u^{(\infty)}\| \leq \varepsilon \|u^{(n)} - u^{(\infty)}\|$ , the above stopping criterion ensures that the distance between our obtained result  $u$  and  $u^{(\infty)}$  is  $\|u - u^{(\infty)}\| \leq \frac{10^{-6}\varepsilon}{1-\varepsilon}$ . Moreover, if we set  $\alpha_2 > 0$ , then we depict the minimal norm of Lagrange multipliers  $\eta^{(n)} := \min_i \{\|\eta_i^{(n)}\|_{\ell^2(\Omega)}\}$ , which – according to Corollary 2.15 (see also Theorem 2.14) – indicates how close the computed solution is to a real global solution. In fact, when the minimal norm of Lagrange multipliers tends to zero numerically, then the associated domain decomposition algorithm converges (along a subsequence) indeed to a global solution; see Figure 2.4(c)-(d), 2.6(c)-(d), and 2.8(c).

We apply the overlapping and nonoverlapping domain decomposition algorithm in (2.9) to the image shown in Figure 2.4(a) by decomposing the image domain into two overlapping or nonoverlapping subdomains. This image of size  $167 \times 270$  pixels has partly lost data (black heart) while it is also corrupted by 10% salt-and-pepper noise (i.e., 10% of the pixels are either flipped to black or white) and by Gaussian white noise with zero mean and variance 0.03. In this example the operators  $T_1$  and  $T_2$  act as  $T_i u = 1_{\Omega \setminus K} u$  for  $i = 1, 2$ , where  $\Omega$  denotes the image domain and  $K \subset \Omega$  the set in which the original image content got lost. The parameters  $\alpha_1$  and  $\alpha_2$  are chosen to be 0.4, while  $\gamma = 0.01$ ,  $\mu = 1$ , and  $\delta = 0.99$ . In Figure 2.4(b) we depict the result computed by the overlapping domain decomposition algorithm. Since  $\alpha_2$  is chosen to be positive, the progress of the minimal norm of the Lagrange multipliers allows us to check whether the iterates converge to a minimizer of the global functional. In fact, we see in Figure 2.4(c) and (d) that the minimal norm of the Lagrange multipliers converges to 0, and hence the accumulation points of the sequence of iterates converge to a global minimizer.

In the next example we present the successful application of a domain decomposition for the problem of pure  $L^1$ -TV minimization. Figure 2.5(a) shows the previously used image rescaled to size  $334 \times 540$  pixels, which is now corrupted by a Gaussian blur with a kernel size of  $15 \times 15$

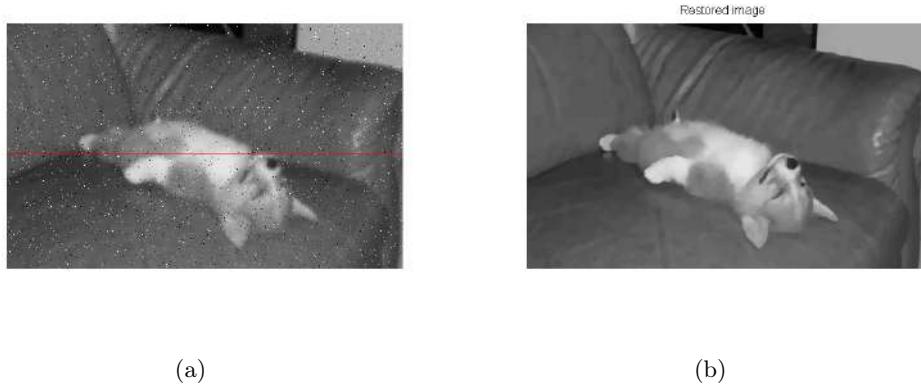
## 2. Subspace Correction for Nonsmooth and Nonadditive Problems



**Figure 2.4:** Domain decomposition for  $L^1$ - $L^2$ -TV minimization. Parameters:  $\alpha_1 = \alpha_2 = 0.4$ ,  $\gamma = 0.01$ ,  $\mu = 1$ ,  $\delta = 0.99$ , and ROF-problem solved via the split Bregman method with tolerance  $10^{-3}$ . In (a) we show an image of size  $167 \times 270$  pixels with a missing part (black heart) and corrupted by 10% salt-and-pepper noise and Gaussian noise with zero mean and variance 0.03. The restored image is shown in (b). In (c) we depict the progress of the minimal norm of Lagrange multipliers  $\eta^{(n)}$  obtained by overlapping domain decomposition, as depicted in (e), where the red and blue lines mark the interfaces of the domain decomposition patches, while in (d) we plot the one obtained by the nonoverlapping domain decomposition; see (f) with highlighted partition.

## I. Subspace Correction Methods for Total Variation Minimization

pixels and standard deviation 2 and also by 2% salt-and-pepper noise. In order to restore the image we decompose the image domain into two nonoverlapping subdomains and solve the resulting problems on the respective subdomains alternatingly by the nonoverlapping domain decomposition algorithm (2.48). Since there is no Gaussian noise present, this is a typical example for  $L^1$ -TV minimization; i.e., we set  $\alpha_2 = 0$  in (2.51). We choose  $\alpha_1 = \frac{5}{3}$ ,  $\gamma = 0.01$ ,  $\mu = 1$ , and  $\delta = 0.99$  and obtain the image in Figure 2.5(b).

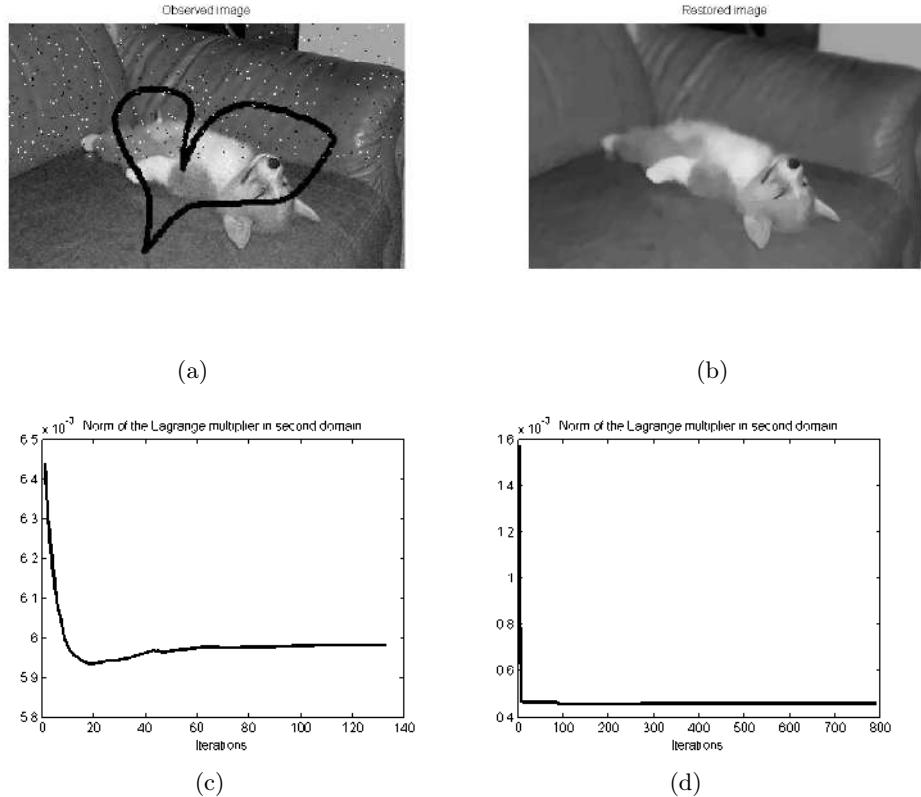


**Figure 2.5:** Nonoverlapping domain decomposition algorithm for  $L^1$ -TV minimization. Parameters:  $\alpha_1 = 5/3$ ,  $\gamma = 0.01$ ,  $\mu = 1$ ,  $\delta = 0.99$ , and ROF-problem solved via the split Bregman method with tolerance  $10^{-4}$ . In (a) we show an image of size  $334 \times 540$  pixels which is corrupted by a Gaussian blur (size  $15 \times 15$ ; standard deviation 2) and 2% salt-and-pepper noise. In this simulation the problem is split into two subproblems. The restored image is shown in (b).

Further we illustrate the successful application of the nonoverlapping domain decomposition algorithm (2.48) when both salt-and-pepper noise and Gaussian noise are present. In particular, we apply our algorithm to an image with a missing part which is corrupted by salt-and-pepper noise in the upper half while in the lower half only Gaussian noise is present. We are aware that this is a rather artificial example but very interesting from a numerical point of view. Note that since the total variation is nonlocal and hence nonadditive, it is not possible to obtain a correct global solution by just cutting the image into an upper and a lower part, computing the solutions separately, and then putting them together. However, since we are in the setting of the special situation of Section 2.5.2, by using our nonoverlapping domain decomposition algorithm in (2.48), we are able to split the image into domains in which only one type of noise is present. Then we have to solve on each domain only either a constrained  $L^1$ -TV minimization problem (cf. (2.49)) or a constrained  $L^2$ -TV minimization problem (cf. (2.50)). These problems are in general easier to solve than the original  $L^1$ - $L^2$ -TV problem. Figure 2.6(a) is such an image (size  $167 \times 270$  pixels), which we restore by the nonoverlapping domain decomposition algorithm (2.48) with  $\mu = 100$  in the upper half and  $\mu = 1$  in the lower half,  $\alpha_1 = \frac{5}{3}$ ,  $\alpha_2 = \frac{50}{3}$ ,  $\gamma = 0.01$ , and  $\delta = 0.99$ . The computed result is shown in Figure 2.6(b). By depicting the minimal norm of Lagrange multipliers  $\eta^{(n)}$ , we check additionally whether the algorithm converges to the right solution. In Figure 2.6(c) we see the progress of the minimal norm of Lagrange multipliers for the image in Figure 2.6(a) with size  $167 \times 270$  pixels. By improving the image resolution threefold, i.e., yielding an image with size  $501 \times 810$  pixels, we

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

obtain a sequence  $(\eta^{(n)})_n$  which converges to a significantly smaller norm; see Figure 2.6(d). If we keep increasing the image resolution, we observe that the sequence  $(\eta^{(n)})_n$  continues to converge to smaller and smaller values. This behavior may be attributed to the fact that the support of  $\eta_i^{(n)}$  is confined to a small stripe of width two pixels, and this depends on the mesh size  $h$ .



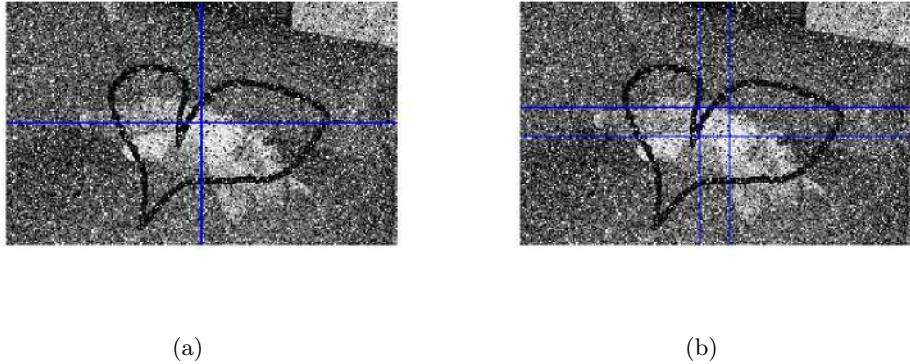
**Figure 2.6:** Nonoverlapping domain decomposition algorithm for  $L^1$ - $L^2$ -TV minimization. Parameters:  $\alpha_1 = \frac{5}{3}$ ,  $\alpha_2 = \frac{50}{3}$ ,  $\gamma = 0.01$ ,  $\mu = 100$  in  $\Omega_1$  and  $\mu = 1$  in  $\Omega_2$ ,  $\delta = 0.99$ , and ROF-problem solved via the split Bregman method with tolerance  $10^{-4}$ . In (a) we show an image of size  $167 \times 270$  pixels with a missing part (black heart) and corrupted by 2% salt-and-pepper noise and Gaussian noise with variance 0.001. The restored image is shown in (b). In (c) we depict the progress of the minimal norm of Lagrange multipliers  $\eta^{(n)}$  as well as in (d) for a three times finer grid.

With respect to the Gauss-Southwell- $r$  rule considered in Section 2.4.2.2, we observe in our previous  $L^1$ - $L^2$ -TV minimization examples that inequality (2.46) is more likely to be satisfied with a fixed constant  $\nu > 0$  for overlapping rather than nonoverlapping domain decomposition. For instance, for the problem depicted in Figure 2.4, the Gauss-Southwell- $r$  rule is satisfied with  $\nu \leq 0.03837$  along the iteration in the overlapping case. For the nonoverlapping decomposition one has  $\nu \leq 0.000195$ .

## I. Subspace Correction Methods for Total Variation Minimization

### Sequential Domain Decomposition Algorithm versus Global $L^1$ - $L^2$ -TV Algorithm

We also compare the performance of the sequential domain decomposition algorithms with the  $L^1$ - $L^2$ -TV algorithm, which solves the considered problems on all of  $\Omega$  without any splitting into subdomains. Since we are comparing now the convergence speed of different algorithms, the stopping criterion used before is no longer suitable. Now we stop the algorithms as soon as the energy  $J_{\alpha_1, \alpha_2}$  reaches a significance level  $J^*$ , i.e., when  $J_{\alpha_1, \alpha_2}(u^{(n)}) \leq J^*$  for the first time. The level  $J^*$  is chosen visually; i.e., we once restore the image of interest until we observe a visually satisfying restoration and record the associated energy value as  $J^*$ . Such a reasonable restoration can be obtained, for example, by running one of the algorithms until  $\|u^{(n)} - u^{(n+1)}\| < 10^{-6}$  for the first time, as it was done above for the domain decomposition algorithms.



**Figure 2.7:** Splitting of the image in Figure 2.4 into four nonoverlapping and overlapping windows.

For our comparison we solved the problem associated with Figure 2.4 by considering splittings into  $D = 2, 4, 8$  overlapping and nonoverlapping stripes, as shown in Figure 2.4, and into  $D = 4$  overlapping and nonoverlapping windows, as depicted in Figure 2.7. The width  $w$  of the overlap is chosen to be 2 or 10 pixels. We stop the algorithms as soon as they reach the significant energy of  $J^* = 0.098973$ . In Table 2.2 we summarize our findings. One clearly observes that the sequential domain decomposition algorithms are much faster than the global  $L^1$ - $L^2$ -TV algorithm. Since the domain decomposition approach considered here is sequential, the convergence slows down in time when the number of subdomains is increased. Nevertheless, the number of iterations is nonincreasing with the number of subdomains, and even decreasing for the overlapping version. Moreover, in Table 2.2 we also present the minimal norm of Lagrange multipliers to indicate that in all of our experiments we are indeed very close to a global minimizer.

A different behaviour can be observed for the deblurring problem of Figure 2.5, where only salt-and-pepper noise is present and therefore  $\alpha_2$  is set to zero. We again tested the sequential domain decomposition algorithm for different splittings ( $D = 2, 4, 8$ ) and compared the performance with the global algorithm; see Table 2.3. Now the sequential domain decomposition approach became computationally more expensive. However, we note that in general one should not expect that the sequential domain decomposition algorithms always outperform the global one. Here this is particularly true, as  $T_1$  is a nonlocal operator and the subproblems

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

# domains	Nonoverlapping alg.	Overlapping alg. (overlap $w = 10$ pixels)	Overlapping alg. (overlap $w = 2$ pixels)
$D = 1$ ( $L^1$ - $L^2$ -TV alg.)	471 s / 285 it / 44.55		
$D = 2$ (stripe)	170 s / 41 it / 45.43 / $2.1 \cdot 10^{-7}$	162 s / 38 it / 45.13 / $1.1 \cdot 10^{-7}$	160 s / 40 it / 45.36 / $9.1 \cdot 10^{-8}$
$D = 4$ (stripe)	215 s / 41 it / 45.41 / $1.5 \cdot 10^{-7}$	225 s / 33 it / 45.18 / $1.3 \cdot 10^{-7}$	234 s / 39 it / 45.40 / $1.0 \cdot 10^{-7}$
$D = 4$ (window)	207 s / 41 it / 45.37 / $7.6 \cdot 10^{-8}$	168 s / 36 it / 45.02 / $8.0 \cdot 10^{-8}$	164 s / 40 it / 45.31 / $6.4 \cdot 10^{-8}$
$D = 8$ (stripe)	285 s / 41 it / 45.33 / $1.1 \cdot 10^{-7}$	213 s / 24 it / 45.01 / $7.8 \cdot 10^{-8}$	248 s / 35 it / 45.25 / $7.0 \cdot 10^{-8}$

**Table 2.2:** Restoration of the image in Figure 2.4: Computational performance (CPU time in seconds / the number of iterations / PSNR-value) for the global  $L^1$ - $L^2$ -TV algorithm and computational performance (CPU time in seconds / the number of iterations / PSNR-value /  $\min\{\|\eta_1\|^2, \dots, \|\eta_D\|^2\}$ ) for the sequential domain decomposition algorithms with  $\alpha_1 = \alpha_2 = 0.4$  for different numbers of subdomains ( $D = 2, 4, 8$ ) and overlapping sizes. The algorithms are stopped as soon as the energy reaches the significance level  $J^* = 0.098973$ .

involve constraints, adding to the complexity of the solution approach.

In the next section we show the successful application of our solvers when both Gaussian noise as well as salt-and-pepper noise are present simultaneously (and in a nonseparated fashion) in an image; see the results depicted in Figure 2.8.

### 2.5.4.2. Numerical Results – Parallel Algorithms

Finally, we show the efficiency of the parallel algorithm in (2.10) for nonoverlapping and overlapping domain decomposition and compare its numerical performance with the  $L^1$ - $L^2$ -TV algorithm introduced in Section 2.2.2. Note that in the  $L^1$ - $L^2$ -TV algorithm the problem is solved on all of  $\Omega$  without any splitting into subdomains. In the domain decomposition algorithms we consider domain splittings into  $D = 4, 8, 16, 32$  subdomains. Since we are comparing the convergence speed of different algorithms, we stop the algorithms as soon as the energy  $J_{\alpha_1, \alpha_2}$  reaches a significance level  $J^*$ , as already described above.

For our comparison let us consider the image in Figure 2.8, which is of size  $1920 \times 2576$  pixels and corrupted by Gaussian noise with standard deviation 0.01 as well as by 10% salt-and-pepper noise on all  $\Omega$ ; see Figure 2.8(a). In the domain decomposition algorithms as well as in the  $L^1$ - $L^2$ -TV algorithm we denoise this image by choosing  $\alpha_1 = 0.5$ , and  $\alpha_2 = 0.4$ . The computations are done in MATLAB on a Linux cluster with 32 kernels, where each kernel has two processors and each processor four cores, i.e., on a computer with 256 cores, and the multithreading-option is activated such that all algorithms (including the  $L^1$ - $L^2$ -TV algorithm without domain decomposition) take advantage of the parallel infrastructure offered by the hardware. For the domain decomposition algorithms we split the domain into nonoverlapping or overlapping strips. The overlap is chosen to be a stripe of width 10 pixels, i.e., the overlap is of size  $10 \times 2576$  pixels. For different numbers of splittings we show in

## I. Subspace Correction Methods for Total Variation Minimization

# domains	Nonoverlapping alg.	Overlapping alg. (overlap $w = 10$ pixels)	Overlapping alg. (overlap $w = 2$ pixels)
$D = 1$ ( $L^1$ -TV alg.)	153 s / 6 it / 82.03		
$D = 2$ (stripe)	566 s / 10 it / 81.18	580 s / 9 it / 81.15	550 s / 9 it / 81.22
$D = 4$ (stripe)	445 s / 10 it / 81.17	484 s / 9 it / 81.02	430 s / 9 it / 81.18
$D = 4$ (window)	393 s / 5 it / 80.26	384 s / 5 it / 80.31	458 s / 5 it / 80.29
$D = 8$ (stripe)	508 s / 10 it / 81.19	467 s / 9 it / 80.97	486 s / 9 it / 81.18

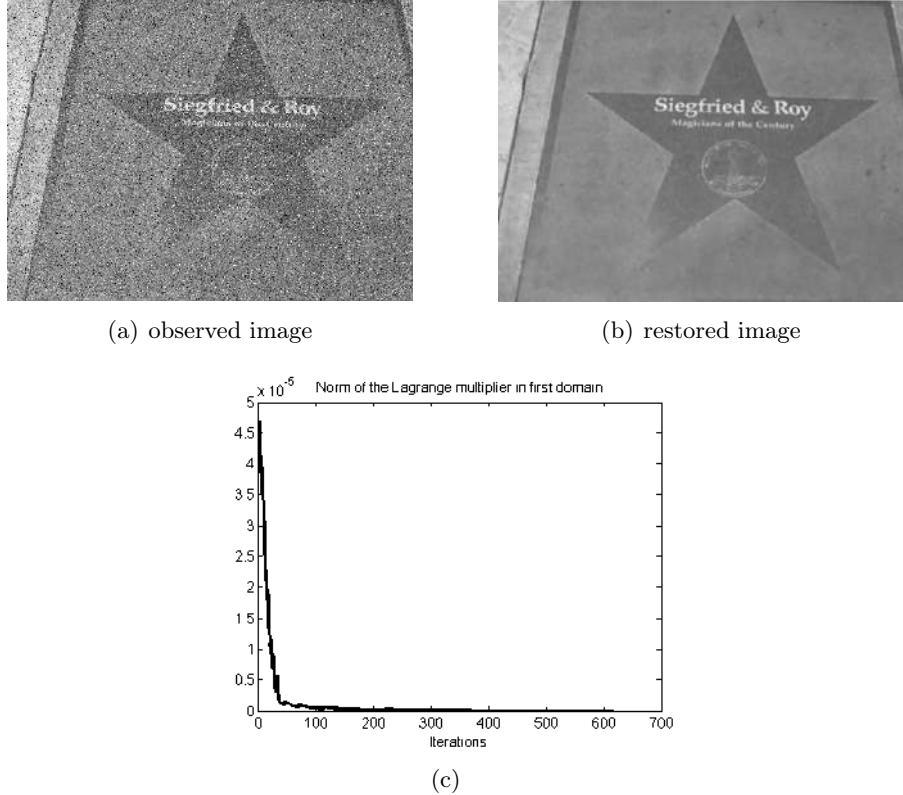
**Table 2.3:** Restoration of the image in Figure 2.5: Computational performance (CPU time in seconds / the number of iterations / PSNR-value) for the global  $L^1$ -TV algorithm and for the sequential domain decomposition algorithms with  $\alpha_1 = 5/3$  for different numbers of subdomains ( $D = 2, 4, 8$ ) and overlapping sizes. The algorithms are stopped as soon as the energy reaches the significance level  $J^* = 0.03052$ .

Table 2.4 the required computational time and the number of iterations until the algorithms reach the significant energy of  $J^* = 0.080041483485$  (see Figure 2.8(b) for the restored image). Note that the structure of the problems in the subdomains is different from the one of the global problem. More precisely, on each subdomain we have to solve constrained minimization problems (cf (2.11)), which are structurally more difficult to solve than just minimizing an energy as for the global problem. Hence by domain decomposition, on the one hand, we reduce the dimensionality of the problem, but, on the other hand, we increase the complexity on each subdomain. Additionally, we also have to take the communication time of the processors into account. These facts add to the overall computing time. Therefore we cannot expect a very dramatic decrease in computational time once the number of subdomains gets large. Nevertheless, we see in Table 2.4 that the domain decomposition algorithms for splittings with  $D = 4, 8, 16, 32$  are still much faster than without decomposition ( $D = 1$ ). In this case, for a nonoverlapping splitting into 8 domains the best performance is guaranteed, while for decomposing into 16 or more domains the algorithm already requires more time to reach its stopping criterion.

Splitting the image domain into larger subdomains, as happens for an overlapping decomposition, one may expect an increase in computational time. This may not necessarily occur, as the solution in the overlap is computed twice per iteration, which decreases the number of iterations. We even see in Table 2.4 that for a fixed number of subdomains the larger the overlapping region is, the fewer iterations are performed. In our numerical experiments we observe that for an overlapping decomposition into 16 domains with overlaps of size  $50 \times 2576$  pixels the domain decomposition algorithm performs best with respect to the number of iterations and computational time.

We also observe that, with increasing number of subdomains, the number of iterations is decreasing. For a nonoverlapping decomposition the number of iterations is only decreasing very slowly, while for overlapping decompositions the decay is more noticeable. For the overlapping splitting, when doubling the number of domains, we see from Table 2.4 that for a larger overlap the absolute reduction of the number of iterations is larger than for a smaller overlap, while the relative reduction is bigger for smaller overlap.

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems



**Figure 2.8:** Parallel domain decomposition for  $L^1$ - $L^2$ -TV minimization of an image (size  $1920 \times 2576$  pixels) corrupted by Gaussian noise with standard deviation 0.01 and 10% salt-and-pepper noise, (a). In (b) we show the image restored using the nonoverlapping domain decomposition algorithm for 8 domains with the following parameters:  $\alpha_1 = 0.5$ ,  $\alpha_2 = 0.4$ ,  $\gamma = 0.01$ ,  $\mu = 10$ ,  $\delta = 0.99$ , and ROF-problem solved via the split Bregman method with tolerance  $10^{-3}$ . In (c) we depict the progress of the minimal norm of Lagrange multipliers  $\eta^{(n)}$ .

# domains	Nonoverlapping alg.	Overlapping alg. (overlap $10 \times 2576$ pixels)	Overlapping alg. (overlap $50 \times 2576$ pixels)
$D = 1$ ( $L^1$ - $L^2$ -TV alg.)	7882 s / 698 it		
$D = 4$	5727 s / 617 it	5834 s / 607 it	5998 s / 561 it
$D = 8$	5090 s / 618 it	5074 s / 596 it	5265 s / 499 it
$D = 16$	5409 s / 588 it	5432 s / 560 it	5014 s / 371 it
$D = 32$	6814 s / 586 it	6605 s / 501 it	5203 s / 242 it

**Table 2.4:** Denoising for the image in Figure 2.8: Computational performance (CPU time in seconds and the number of iterations) for the global  $L^1$ - $L^2$ -TV algorithm and for the parallel domain decomposition algorithms with  $\alpha_1 = 0.5$ ,  $\alpha_2 = 0.4$  for different numbers of subdomains ( $D = 4, 8, 16, 32$ ) and overlapping sizes.

## I. Subspace Correction Methods for Total Variation Minimization

### 2.6. Conclusion

We have proposed a combined  $L^1$ - $L^2$ -TV functional with total variation regularization which outperforms the pure  $L^1$ -TV or  $L^2$ -TV models, as it preserves details better than the  $L^2$ -TV model and does not suffer from a sudden loss of image features like the  $L^1$ -TV model. Moreover, it is superior (in PSNR) in restoration tasks where images are corrupted simultaneously by Gaussian and salt-and-pepper noise.

For the numerical solution of the  $L^1$ - $L^2$ -TV energy we have proposed and analyzed sequential and parallel subspace correction methods, which generate a convergent (sub)sequence of iterates and a monotone decrease of the energy. Moreover, we have shown that the distance between limit points and the global minimizer of the  $L^1$ - $L^2$ -TV energy is bounded by the minimal norm of Lagrange multipliers associated with involved subspace projection constraints. In our numerical experiments this norm often appeared very small, or it might even tend to 0, indicating that convergence to the global optimum was obtained. However, in the rare cases where the minimal norm of Lagrange multipliers did not tend to 0, we observed a resolution dependent effect reducing the multiplier norm under increasing resolution. This behavior may certainly motivate further research on the convergence of subspace correction methods for minimizing nonsmooth and nonadditive objectives. We have also shown that the parallel version pays off up to the number of subdomains where the communication between processors becomes dominant.

We also mention that the theoretical analysis of subspace correction methods for nonsmooth and nonadditive functionals is still far from being complete. In particular, in general Banach spaces there is not much known about such methods and their convergence to a global minimizer. Not even in a discrete setting for dimensions  $d > 1$  has this question yet been answered without invoking (rather restrictive) assumptions.

### 2.7. References

- [1] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE Transactions on Signal Processing*, 45(4):913–917, 1997.
- [2] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, 2000.
- [3] H. Attouch, G. Buttazzo, and G. Michaille. *Variational Analysis in Sobolev and BV Spaces*. MOS-SIAM Series on Optimization. SIAM, second edition, Philadelphia, PA, 2014. Applications to PDEs and optimization.
- [4] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition – modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67(1):111–136, 2006.
- [5] D. P. Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.
- [6] D. P. Bertsekas, A. Nedić, and A. E. Ozdaglar. Convex Analysis and Optimization. 2003.
- [7] A. Braides.  $\Gamma$ -convergence for Beginners, volume 22 of *Oxford Lecture Series in Mathematics and its Applications*. Oxford University Press, Oxford, 2002.
- [8] L. M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7(3):200–217, 1967.

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

- [9] J.-F. Cai, R. H. Chan, and M. Nikolova. Two-phase approach for deblurring images corrupted by impulse plus Gaussian noise. *Inverse Problems and Imaging*, 2(2):187–204, 2008.
- [10] C. Carstensen. Domain decomposition for a non-smooth convex minimization problem and its application to plasticity. *Numerical Linear Algebra with Applications*, 4(3):177–190, 1997.
- [11] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging Vision*, 20(1-2):89–97, 2004.
- [12] A. Chambolle and J. Darbon. On total variation minimization and surface evolution using parametric maximum flows. *International Journal of Computer Vision*, 84(3):288–307, 2009.
- [13] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, 1997.
- [14] R. H. Chan, Y. Dong, and M. Hintermüller. An efficient two-phase-TV method for restoring blurred images with impulse noise. *IEEE Transactions on Image Processing*, 19(7):1731–1739, 2010.
- [15] R. H. Chan, C.-W. Ho, and M. Nikolova. Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. *IEEE Transactions on Image Processing*, 14(10):1479–1485, 2005.
- [16] T. F. Chan and S. Esedoḡlu. Aspects of total variation regularized  $L^1$  function approximation. *SIAM Journal on Applied Mathematics*, 65(5):1817–1837, 2005.
- [17] T. F. Chan, G. H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.
- [18] T. F. Chan and T. P. Mathew. Domain decomposition algorithms. *Acta Numerica*, 3:61–143, 1994.
- [19] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.
- [20] J. Darbon and M. Sigelle. A fast and exact algorithm for total variation minimization, in *Proceedings of the IbPRIA 2005, Lecture Notes in Computer Science*, 3522: 351–359, Springer, 2005.
- [21] J. Darbon and M. Sigelle. Image restoration with discrete constrained total variation. I. Fast and exact optimization. *Journal of Mathematical Imaging Vision*, 26(3):261–276, 2006.
- [22] I. Daubechies, G. Teschke, and L. Vese. Iteratively solving linear inverse problems under general convex constraints. *Inverse Problems and Imaging*, 1(1):29–46, 2007.
- [23] J. Delon and A. Desolneux. A patch-based approach for removing impulse or mixed Gaussian-impulse noise. *SIAM Journal on Imaging Sciences*, 6(2):1140–1174, 2013.
- [24] D. C. Dobson and C. R. Vogel. Convergence of an iterative method for total variation denoising. *SIAM Journal on Numerical Analysis*, 34(5):1779–1791, 1997.
- [25] Y. Dong, M. Hintermüller, and M. Neri. An efficient primal-dual method for  $L^1$  TV image restoration. *SIAM Journal on Imaging Sciences*, 2(4):1168–1189, 2009.
- [26] I. Ekeland and R. Témam. *Convex Analysis and Variational Problems*, volume 28 of *Classics in Applied Mathematics*. SIAM, Philadelphia, PA, 1999.
- [27] M. Fornasier. Domain decomposition methods for linear inverse problems with sparsity constraints. *Inverse Problems*, 23(6):2505, 2007.
- [28] M. Fornasier, Y. Kim, A. Langer, and C.-B. Schönlieb. Wavelet decomposition method for  $L_2$ /TV-image deblurring. *SIAM Journal on Imaging Sciences*, 5(3):857–885, 2012.
- [29] M. Fornasier, A. Langer, and C.-B. Schönlieb. A convergent overlapping domain decomposition method for total variation minimization. *Numerische Mathematik*, 116(4):645–685, 2010.

## I. Subspace Correction Methods for Total Variation Minimization

- [30] M. Fornasier and C.-B. Schönlieb. Subspace correction methods for total variation and  $l_1$ -minimization. *SIAM Journal on Numerical Analysis*, 47(5):3397–3428, 2009.
- [31] R. Garnett, T. Huegerich, C. Chui, and W. He. A universal noise removal algorithm with an impulse detector. *IEEE Transactions on Image Processing*, 14(11):1747–1754, 2005.
- [32] T. Goldstein and S. Osher. The split Bregman method for  $L_1$ -regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.
- [33] M. Hintermüller and K. Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM Journal on Applied Mathematics*, 64(4):1311–1333, 2004.
- [34] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.
- [35] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms. I*, volume 305 of *Grundlehren der Mathematischen Wissenschaften*. Springer, Berlin, 1993.
- [36] K. Ito and K. Kunisch. *Lagrange multiplier approach to variational problems and applications*, volume 15 of *Advances in Design and Control*. SIAM, Philadelphia, PA, 2008.
- [37] A. Langer. *Subspace Correction and Domain Decomposition Methods for Total Variation Minimization*. PhD thesis, Johannes Kepler Universität Linz, Linz, Austria, 2011.
- [38] A. Langer, S. Osher, and C.-B. Schönlieb. Bregmanized domain decomposition for image restoration. *Journal of Scientific Computing*, 54(2-3):549–576, 2013.
- [39] Y. Meyer. *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations: the Fifteenth Dean Jacqueline B. Lewis Memorial Lectures*, volume 22. AMS, Providence, RI, 2001.
- [40] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming Series A*, 103:127–152, 2005.
- [41] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40(3):965–994, 2002.
- [42] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1-2):99–120, 2004.
- [43] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.
- [44] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, N.J., 1970.
- [45] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.
- [46] X.-C. Tai and P. Tseng. Convergence rate analysis of an asynchronous space decomposition method for convex minimization. *Mathematics of Computation*, 71(239):1105–1135, 2002.
- [47] X.-C. Tai and J. Xu. Global and uniform convergence of subspace correction methods for some convex optimization problems. *Mathematics of Computation*, 71(237):105–124, 2002.
- [48] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications*, 109(3):475–494, 2001.
- [49] P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117(1):387–423, 2009.
- [50] L. Vese. A study in the BV space of a denoising-deblurring variational problem. *Applied Mathematics & Optimization*, 44(2):131–161, 2001.

## 2. Subspace Correction for Nonsmooth and Nonadditive Problems

- [51] C. Vonesch and M. Unser. A fast multilevel algorithm for wavelet-regularized image restoration. *IEEE Transactions on Image Processing*, 18(3):509–523, 2009.
- [52] J. Warga. Minimizing certain convex functions. *Journal of the Society for Industrial and Applied Mathematics*, 11:588–593, 1963.
- [53] P. Weiss, L. Blanc-Féraud, and G. Aubert. Efficient schemes for total variation minimization under constraints in image processing. *SIAM Journal on Scientific Computing*, 31(3):2047–2080, 2009.
- [54] J. Xu. The method of subspace corrections. *Journal of Computational and Applied Mathematics*, 128(1):335–362, 2001.
- [55] W. Yin, S. Osher, D. Goldfarb, and J. Darbon. Bregman iterative algorithms for  $\ell_1$ -minimization with applications to compressed sensing. *SIAM Journal on Imaging Sciences*, 1(1):143–168, 2008.
- [56] X. Zhang, M. Burger, X. Bresson, and S. Osher. Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM Journal on Imaging Sciences*, 3(3):253–276, 2010.
- [57] M. Zhu and T. Chan. An efficient primal-dual hybrid gradient algorithm for total variation image restoration. *UCLA CAM Report*, pages 08–34, 2008.



### 3. Surrogate Functional Based Subspace Correction Methods for Image Processing

**Bibliographic Note:** The content of this chapter is published in the article [AL2]: M. Hintermüller and A. Langer. Surrogate functional based subspace correction methods for image processing. In *Domain Decomposition Methods in Science and Engineering XXI*, pages 829–837, Springer, 2014.

**Summary:** We present subspace correction methods for nonsmooth and nonadditive energies which are guaranteed to converge. Moreover, we are able to provide an estimate of the distance between the outcome of the subspace correction method and the global minimizer of the nonsmooth and nonadditive objective. With the help of this estimate we can finally show in our numerical experiments that the proposed method even converges to a true global minimizer.

#### 3.1. Introduction

Recently in [4, 5, 6] subspace correction methods for nonsmooth and nonadditive problems have been introduced in the context of image processing, where the nonsmooth and nonadditive total variation (TV) plays a fundamental role as a regularization technique, since it preserves edges and discontinuities in images. We recall, that for  $u \in L^1(\Omega)$ ,

$$V(u, \Omega) := \sup \left\{ \int_{\Omega} u \operatorname{div} \phi dx : \phi \in [C_c^1(\Omega)]^2, \|\phi\|_{\infty} \leq 1 \right\}$$

is the variation of  $u$ . In the event that  $V(u, \Omega) < \infty$  we denote  $|Du|(\Omega) = V(u, \Omega)$  and call it the total variation of  $u$  in  $\Omega$  [1].

In this paper, as in [6], we consider functionals, which consist of a nonsmooth and nonadditive regularization term and a weighted combination of an  $\ell^1$ -term and a quadratic  $\ell^2$ -term; see (3.1) below. This type of functional has been shown to be particularly efficient to eliminate simultaneously Gaussian and salt-and-pepper noise. In [6] an estimate of the distance of the limit point obtained from the proposed subspace correction method to the global minimizer is established. In that paper the exact subspace minimization problems are minimized, which are in general not easily solved. Therefore, in the present paper we analyse a subspace correction approach in which the subproblems are approximated by so-called *surrogate* functionals, as in [4, 5]. In this situation, as in [6], we are able to achieve an estimate for the distance of the computed solution to the real global minimizer. With the help of this estimate we show in our numerical experiments that the proposed algorithm generates a sequence which converges to the expected minimizer.

## I. Subspace Correction Methods for Total Variation Minimization

### 3.2. Notations

For the sake of brevity we consider a two dimensional setting only. We define  $\Omega = \{x_1 < \dots < x_N\} \times \{y_1 < \dots < y_N\} \subset \mathbb{R}^2$ , and  $H = \mathbb{R}^{N \times N}$ , where  $N \in \mathbb{N}$ . For  $u \in H$  we write  $u = u(x) = u(x_i, y_j)$ , where  $i, j \in \{1, \dots, N\}$  and  $x \in \Omega$ . Let  $h = x_{i+1} - x_i = y_{j+1} - y_j$  be the equidistant step-size. We define the scalar product of  $u, v \in H$  by  $\langle u, v \rangle_H = h^2 \sum_{x \in \Omega} u(x)v(x)$  and the scalar product of  $p, q \in H^2$  by  $\langle p, q \rangle_{H^2} = h^2 \sum_{x \in \Omega} \langle p(x), q(x) \rangle_{\mathbb{R}^2}$  with  $\langle z, w \rangle_{\mathbb{R}^2} = \sum_{j=1}^2 z_j w_j$  for every  $z = (z_1, z_2) \in \mathbb{R}^2$  and  $w = (w_1, w_2) \in \mathbb{R}^2$ . We also use  $\|u\|_{\ell^p(\Omega)} = (h^2 \sum_{x \in \Omega} |u(x)|^p)^{1/p}$ ,  $1 \leq p < \infty$ ,  $\|u\|_{\ell^\infty(\Omega)} = \sup_{x \in \Omega} |u(x)|$  and  $\|\cdot\|$ , when any norm can be taken.

The discrete gradient  $\nabla u$  is denoted by  $(\nabla u)(x) = ((\nabla u)^1(x), (\nabla u)^2(x))$  with  $(\nabla u)^1(x) = \frac{1}{h}(u(x_{i+1}, y_j) - u(x_i, y_j))$  if  $i < N$  and  $(\nabla u)^1(x) = 0$  if  $i = N$ , and  $(\nabla u)^2(x) = \frac{1}{h}(u(x_i, y_{j+1}) - u(x_i, y_j))$  if  $j < N$  and  $(\nabla u)^2(x) = 0$  if  $j = N$ , for all  $x \in \Omega$ . For  $\omega \in H^2$  we define  $\varphi : \mathbb{R} \rightarrow \mathbb{R}$  by  $\varphi(|\omega|)(\Omega) := h^2 \sum_{x \in \Omega} \varphi(|\omega(x)|)$ , where  $|z| = \sqrt{z_1^2 + z_2^2}$ . In particular we define the *total variation* of  $u$  by setting  $\varphi(t) = t$  and  $\omega = \nabla u$ , i.e.,  $|\nabla u|(\Omega) := h^2 \sum_{x \in \Omega} |\nabla u(x)|$ .

For an operator  $T$  we denote by  $T^*$  its adjoint. Further we introduce the *discrete divergence*  $\operatorname{div} : H^2 \rightarrow H$  defined by  $\operatorname{div} = -\nabla^*$  ( $\nabla^*$  is the adjoint of the gradient  $\nabla$ ), in analogy to the continuous setting. The symbol 1 indicates the constant vector with entry values 1 and  $1_D$  is the characteristic function of  $D \subset \Omega$ .

For a convex functional  $J : H \rightarrow \bar{\mathbb{R}}$ , we define the *subdifferential* of  $J$  at  $v \in H$  as the set valued mapping  $\partial J(v) := \emptyset$  if  $J(v) = \infty$  and  $\partial J(v) := \{v^* \in H : \langle v^*, u - v \rangle_H + J(v) \leq J(u) \quad \forall u \in H\}$  otherwise. It is clear from this definition that  $0 \in \partial J(v)$  if and only if  $v$  is a minimizer of  $J$ . Whenever the underlying space is important, then we write  $\partial_{V_i} J$  or  $\partial_H J$ .

### 3.3. Subspace Correction Approaches

As in [6] we are interested in minimizing by means of subspace correction the following functional

$$J(u) = \alpha_S \|Su - g_S\|_{\ell^1(\Omega)} + \alpha_T \|Tu - g_T\|_{\ell^2(\Omega)}^2 + \varphi(|\nabla u|)(\Omega), \quad (3.1)$$

where  $S, T : H \rightarrow H$  are bounded linear operators,  $g_S, g_T \in H$  are given data, and  $\alpha_S, \alpha_T \geq 0$  with  $\alpha_S + \alpha_T \geq \tau > 0$ . We assume that  $J$  is bounded from below and coercive, i.e.,  $\{u \in H : J(u) \leq C\}$  is bounded in  $H$  for all constants  $C > 0$ , in order to guarantee that (3.1) has minimizers. Moreover we assume that  $\varphi : \mathbb{R} \rightarrow \mathbb{R}$  is a convex function, nondecreasing in  $\mathbb{R}^+$  with (i)  $\varphi(0) = 0$  and (ii)  $cz - b \leq \varphi(z) \leq cz + b$ , for all  $z \in \mathbb{R}^+$  for some constant  $c > 0$  and  $b \geq 0$ .

Note that for the particular example  $\varphi(t) = t$ , the third term in (3.1) becomes the well-known total variation of  $u$  in  $\Omega$  and we call (3.1) the  $L^1$ - $L^2$ -TV model.

Now we seek to minimize (3.1) by decomposing  $H$  into two subspaces  $V_1$  and  $V_2$  such that  $H = V_1 + V_2$ . Note that a generalization to multiple splittings can be performed straightforwardly. However, here we will restrict ourselves to a decomposition into two domains only for simplicity. By  $V_i^c$  we denote the orthogonal complement of  $V_i$  in  $H$  and we define by  $\pi_{V_i^c}$  the corresponding orthogonal projection onto  $V_i^c$  for  $i = 1, 2$ .

With this splitting we want to minimize  $J$  by suitable instances of the following alternating algorithm:

### 3. Surrogate Functional Based Subspace Correction

Choose an initial  $u^{(0)} =: u_1^{(0)} + u_2^{(0)} \in V_1 + V_2$ , for example,  $u^{(0)} = 0$ , and iterate

$$\begin{cases} u_1^{(n+1)} = \arg \min_{u_1 \in V_1} J(u_1 + u_2^{(n)}), \\ u_2^{(n+1)} = \arg \min_{u_2 \in V_2} J(u_1^{(n+1)} + u_2), \\ u^{(n+1)} := u_1^{(n+1)} + u_2^{(n+1)}. \end{cases} \quad (3.2)$$

Differently from the case in [6], where the authors solved the exact subspace minimization problems in (3.2), we suggest now to approximate the subdomain problems by so-called surrogate functionals (cf. [2, 3, 4, 5, 8]): Assume  $a, u_i \in V_i$ ,  $u_{-i} \in V_{-i}$ , and define

$$\begin{aligned} J^s(u_i + u_{-i}, a + u_{-i}) &:= J(u_i + u_{-i}) + \alpha_T (\delta \|u_i + u_{-i} - (a + u_{-i})\|_{\ell^2(\Omega)}^2 \\ &\quad - \|T(u_i + u_{-i} - (a + u_{-i}))\|_{\ell^2(\Omega)}^2) \\ &= J(u_i + u_{-i}) + \alpha_T (\delta \|u_i - a\|_{\ell^2(\Omega)}^2 - \|T(u_i - a)\|_{\ell^2(\Omega)}^2) \end{aligned} \quad (3.3)$$

for  $i = 1, 2$  and  $-i \in \{1, 2\} \setminus \{i\}$ , where  $\delta > \|T\|^2$ . Then an approximate solution to  $\min_{u_i \in V_i} J(u_1 + u_2)$  is realized by using the following algorithm: For  $u_i^{(0)} \in V_i$ ,

$$u_i^{(\ell+1)} = \arg \min_{u_i \in V_i} J^s(u_i + u_{-i}, u_i^{(\ell)} + u_{-i}), \quad \ell \geq 0,$$

where  $u_{-i} \in V_{-i}$  for  $i = 1, 2$  and  $-i \in \{1, 2\} \setminus \{i\}$ .

The alternating domain decomposition algorithm reads then as follows:

Choose an initial  $u^{(0)} =: \tilde{u}_1^{(0)} + \tilde{u}_2^{(0)} \in V_1 + V_2$ , for example,  $u^{(0)} = 0$ , and iterate

$$\begin{cases} \begin{cases} u_1^{(n+1,0)} = \tilde{u}_1^{(n)}, \\ u_1^{(n+1,\ell+1)} = \arg \min_{u_1 \in V_1} J^s(u_1 + \tilde{u}_2^{(n)}, u_1^{(n+1,\ell)} + \tilde{u}_2^{(n)}), \end{cases} \quad \ell = 0, \dots, L-1, \\ \begin{cases} u_2^{(n+1,0)} = \tilde{u}_2^{(n)}, \\ u_2^{(n+1,m+1)} = \arg \min_{u_2 \in V_2} J^s(u_1^{(n+1,L)} + u_2, u_2^{(n+1,m)} + u_1^{(n+1,L)}), \end{cases} \quad m = 0, \dots, M-1, \\ u^{(n+1)} := u_1^{(n+1,L)} + u_2^{(n+1,M)}, \quad \tilde{u}_1^{(n+1)} = \chi_1 \cdot u^{(n+1)}, \quad \tilde{u}_2^{(n+1)} = \chi_2 \cdot u^{(n+1)}, \end{cases} \quad (3.4)$$

where  $\chi_1, \chi_2 \in H$  have the properties (i)  $\chi_1 + \chi_2 = 1$  and (ii)  $\chi_i \in V_i$  for  $i = 1, 2$ . Let  $\kappa := \max\{\|\chi_1\|_\infty, \|\chi_2\|_\infty\} < \infty$ .

The parallel version of the algorithm in (3.4) reads as follows:

Choose an initial  $u^{(0)} =: \tilde{u}_1^{(0)} + \tilde{u}_2^{(0)} \in V_1 + V_2$ , for example,  $u^{(0)} = 0$ , and iterate

$$\begin{cases} \begin{cases} u_1^{(n+1,0)} = \tilde{u}_1^{(n)}, \\ u_1^{(n+1,\ell+1)} = \arg \min_{u_1 \in V_1} J^s(u_1 + \tilde{u}_2^{(n)}, u_1^{(n+1,\ell)} + \tilde{u}_2^{(n)}), \end{cases} \quad \ell = 0, \dots, L-1, \\ \begin{cases} u_2^{(n+1,0)} = \tilde{u}_2^{(n)}, \\ u_2^{(n+1,m+1)} = \arg \min_{u_2 \in V_2} J^s(\tilde{u}_1^{(n)} + u_2, u_2^{(n+1,m)} + \tilde{u}_1^{(n)}), \end{cases} \quad m = 0, \dots, M-1, \\ u^{(n+1)} := \frac{u_1^{(n+1,L)} + u_2^{(n+1,M)} + u^{(n)}}{2}, \quad \tilde{u}_1^{(n+1)} = \chi_1 \cdot u^{(n+1)}, \quad \tilde{u}_2^{(n+1)} = \chi_2 \cdot u^{(n+1)}. \end{cases} \quad (3.5)$$

Note that we prescribe a finite number  $L$  and  $M$  of inner iterations for each subspace, respectively. Hence we do not get a minimizer of the original subspace minimization problems

## I. Subspace Correction Methods for Total Variation Minimization

in (3.2), but approximations of such minimizers. Moreover, observe that  $u^{(n+1)} = \tilde{u}_1^{(n+1)} + \tilde{u}_2^{(n+1)}$ , with  $\tilde{u}_i^{(n+1,L)} \neq \tilde{u}_i^{(n+1)}$ , for  $i = 1, 2$ , in general.

We have that  $u_1^{(n+1,L)} \in \arg \min_{u \in H} \left\{ J^s(u + \tilde{u}_2^{(n)}, u_1^{(n+1,L-1)} + \tilde{u}_2^{(n)}) : \pi_{V_1^c} u = 0 \right\}$ . Then, by [7, Theorem 2.1.4, p. 305] there exists an  $\eta_1^{(n+1)} \in \text{Range}(\pi_{V_1^c})^* \simeq V_1^c$  such that

$$0 \in \partial_H J^s(\cdot + \tilde{u}_2^{(n)}, u_1^{(n+1,L-1)} + \tilde{u}_2^{(n)})(u_1^{(n+1,L)}) + \eta_1^{(n+1)}. \quad (3.6)$$

Analogously, we have that if  $u_2^{(n+1,M)}$  is a minimizer of the second optimization problem in (3.4) or (3.5), then there exists an  $\eta_2^{(n+1)} \in \text{Range}(\pi_{V_2^c})^* \simeq V_2^c$  such that

$$0 \in \partial_H J^s(u_1^{(n+1,L)} + \cdot, u_1^{(n+1,L)} + \tilde{u}_2^{(n+1,M-1)})(u_2^{(n+1,M)}) + \eta_2^{(n+1)}, \text{ or} \quad (3.7)$$

$$0 \in \partial_H J^s(\tilde{u}_1^{(n,L)} + \cdot, \tilde{u}_1^{(n,L)} + \tilde{u}_2^{(n+1,M-1)})(u_2^{(n+1,M)}) + \eta_2^{(n+1)}, \quad (3.8)$$

respectively.

### 3.3.1. Convergence Properties

In this section we state convergence properties of the subspace correction methods in (3.4) and (3.5). In particular, the following three propositions are direct consequences of statements in [4, 5, 6].

**Proposition 3.1.** *The algorithms in (3.4) and (3.5) produce a sequence  $(u^{(n)})_n$  in  $H$  with the following properties:*

1.  $J(u^{(n)}) > J(u^{(n+1)})$  for all  $n \in \mathbb{N}$  (unless  $u^{(n)} = u^{(n+1)}$ );
2.  $\lim_{n \rightarrow \infty} \|u_1^{(n+1,\ell+1)} - u_1^{(n+1,\ell)}\|_{\ell^2(\Omega)} = 0$  and  $\lim_{n \rightarrow \infty} \|u_2^{(n+1,m+1)} - u_2^{(n+1,m)}\|_{\ell^2(\Omega)} = 0$  for all  $\ell \in \{0, \dots, L-1\}$  and  $m \in \{0, \dots, M-1\}$ ;
3.  $\lim_{n \rightarrow \infty} \|u^{(n+1)} - u^{(n)}\|_{\ell^2(\Omega)} = 0$ ;
4. the sequence  $(u^{(n)})_n$  has subsequences that converge in  $H$ .

The proof of this proposition is analogous to the one in [5, Theorem 5.1].

**Proposition 3.2.** *The sequences  $(\tilde{u}_i^{(n)})_n$  for  $i = 1, 2$  generated by the algorithm in (3.4) or (3.5) are bounded in  $H$  and hence have accumulation points  $\tilde{u}_i^{(\infty)}$ , respectively.*

*Proof.* By the boundedness of the sequence  $(u^{(n)})_n$  we obtain  $\|\tilde{u}_i^{(n)}\| = \|\chi_i u^{(n)}\| \leq \kappa \|u^{(n)}\| \leq C < \infty$  and hence  $(\tilde{u}_i^{(n)})_n$  is bounded for  $i = 1, 2$ .  $\square$   $\square$

**Remark 3.3.** *From the previous proposition it directly follows by the coercivity assumption on  $J$  that the sequences  $(u_1^{(n,\ell)})_n$  and  $(u_2^{(n,m)})_n$  are bounded for all  $\ell \in \{0, \dots, L\}$  and  $m \in \{0, \dots, M\}$ .*

**Proposition 3.4.** *Let  $u_1^{(\infty)}$ ,  $u_2^{(\infty)}$ , and  $\tilde{u}_i^{(\infty)}$  be accumulation points of the sequences  $(u_1^{(n,L)})_n$ ,  $(u_2^{(n,M)})_n$ , and  $(\tilde{u}_i^{(n)})_n$  generated by the algorithms in (3.4) and (3.5), then  $u_i^{(\infty)} = \tilde{u}_i^{(\infty)}$ , for  $i = 1, 2$ .*

### 3. Surrogate Functional Based Subspace Correction

One shows this statement analogous to the first part of the proof of [4, Theorem 5.7].

Moreover, as in [6] we are able to establish an estimate of the distance of the limit point obtained from the subspace correction method to the true global minimizer.

**Theorem 3.5.** *Let  $\alpha_S \geq \tau$ ,  $u^*$  a minimizer of  $J$ , and  $u^{(\infty)}$  an accumulation point of the sequence  $(u^{(n)})_n$  generated by the algorithm in (3.4) or (3.5). Then we have that*

1.  $u^{(\infty)}$  is a minimizer of  $J$  or
2. there exists a constant  $\beta > 0$  (independent of  $\alpha_T$ ) such that  $\|u^{(\infty)} - u^*\|_{\ell^2(\Omega)} \leq \beta$  or
3. if  $\alpha_T < \frac{\gamma}{\beta^2 \delta}$  for  $0 < \gamma \leq J(u^{(\infty)}) - J(u^*)$ , then  $\|u^{(\infty)} - u^*\|_{\ell^2(\Omega)} \leq \frac{\beta^2 \|\hat{\eta}\|_{\ell^2(\Omega)}}{\gamma - \alpha_T \delta \beta^2}$ , where  $\|\hat{\eta}\|_{\ell^2(\Omega)} = \min\{\|\eta_1^{(\infty)}\|_{\ell^2(\Omega)}, \|\eta_2^{(\infty)}\|_{\ell^2(\Omega)}\}$  and  $\eta_i^{(\infty)}$  is an accumulation point of the sequence  $(\eta_i^{(n)})_n$  for  $i = 1, 2$  defined as in (3.6)-(3.8) respectively, or
4. if  $T^*T$  is positive definite with smallest Eigenvalue  $\sigma > 0$ ,  $\alpha_T > 0$  and  $\|T\|^2 < \delta < 2\sigma$ , then we have  $\|u^* - u^{(\infty)}\|_{\ell^2(\Omega)} \leq \frac{\|\hat{\eta}\|_{\ell^2(\Omega)}}{\alpha_T(2\sigma - \delta)}$ .

*Proof.* Since  $(u_1^{(n+1,L)})_n$ ,  $(u_1^{(n+1,L-1)})_n$ , and  $(\tilde{u}_2^{(n)})_n$  are bounded and based on the fact that  $\partial J^s(\xi, \tilde{\xi})$  is compact for any  $\xi, \tilde{\xi} \in H$  we obtain that  $(\eta_1^{(n)})_n$  is bounded, cf. [6, Corollary 4.7]. By noting that  $(u_1^{(n+1,L)})_n$  and  $(u_1^{(n+1,L-1)})_n$  have the same limit for  $n \rightarrow \infty$ , see Proposition 3.1, we subtract a suitable subsequence  $(n_k)_k$  with limits  $\eta_1^{(\infty)}$ ,  $u_1^{(\infty)}$ , and  $\tilde{u}_2^{(\infty)}$  such that (3.6)-(3.8) respectively are still valid, cf. [9, Theorem 24.4, p 233], i.e.,  $0 \in \partial_H J^s(\cdot + \tilde{u}_2^{(\infty)}, u_1^{(\infty)} + \tilde{u}_2^{(\infty)})(u_1^{(\infty)}) + \eta_1^{(\infty)}$ . By the definition of the subdifferential and Proposition 3.4 we obtain  $J(u^{(\infty)}) = J^s(u^{(\infty)}, u^{(\infty)}) \leq J^s(v, u^{(\infty)}) + \langle \eta_1^{(\infty)}, u^{(\infty)} - v \rangle_H \leq J^s(v, u^{(\infty)}) + \|\eta_1^{(\infty)}\|_{\ell^2(\Omega)} \|u^{(\infty)} - v\|_{\ell^2(\Omega)}$  for all  $v \in H$ . Similarly one can show that  $J(u^{(\infty)}) \leq J^s(v, u^{(\infty)}) + \|\eta_2^{(\infty)}\|_{\ell^2(\Omega)} \|u^{(\infty)} - v\|_{\ell^2(\Omega)}$  for all  $v \in H$ , and hence we have

$$J(u^{(\infty)}) \leq J^s(v, u^{(\infty)}) + \|\hat{\eta}\|_{\ell^2(\Omega)} \|u^{(\infty)} - v\|_{\ell^2(\Omega)} \quad (3.9)$$

for all  $v \in H$ , where  $\|\hat{\eta}\|_{\ell^2(\Omega)} = \min\{\|\eta_1^{(\infty)}\|_{\ell^2(\Omega)}, \|\eta_2^{(\infty)}\|_{\ell^2(\Omega)}\}$ .

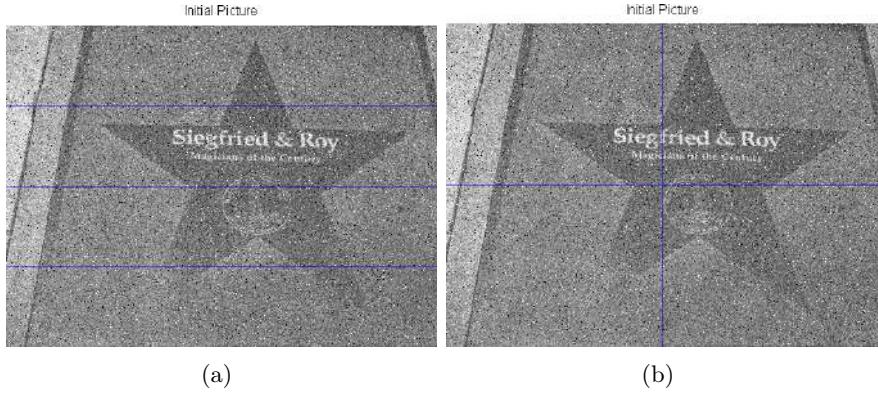
Let  $u^* \in \arg \min_{u \in H} J(u)$ . Then there exists a  $\rho \geq 0$  such that  $J(u^{(\infty)}) = J(u^*) + \rho$ .

1. If  $\rho = 0$ , then it immediately follows that  $u^{(\infty)}$  is a minimizer of  $J$ .
2. If  $\rho > 0$ , then from the coercivity condition we obtain that there exists a constant  $\beta > 0$ , independent of  $\alpha_T$ , such that  $\|u^{(\infty)} - u^*\|_{\ell^2(\Omega)} \leq \beta < +\infty$ .
3. If  $\alpha_T < \frac{\gamma}{\beta^2 \delta}$  for  $0 < \gamma \leq J(u^{(\infty)}) - J(u^*)$ , then  $J(u^{(\infty)}) \geq J(u^*) + \frac{\gamma}{\beta^2} \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}^2$ . Setting  $v = u^*$  in (3.9) and using the last inequality we obtain

$$\begin{aligned} \alpha_T \left( \delta \|u^* - u^{(\infty)}\|_{\ell^2(\Omega)}^2 - \|T(u^* - u^{(\infty)})\|_{\ell^2(\Omega)}^2 \right) &+ \|\hat{\eta}\|_{\ell^2(\Omega)} \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)} \\ &\geq \frac{\gamma}{\beta^2} \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}^2. \end{aligned} \quad (3.10)$$

From the latter inequality we get  $\|\hat{\eta}\|_2 \geq (\frac{\gamma}{\beta^2} - \alpha_T \delta) \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}$  and since  $\alpha_T \delta < \frac{\gamma}{\beta^2}$  we obtain  $\frac{\beta^2 \|\hat{\eta}\|_{\ell^2(\Omega)}}{\gamma - \alpha_T \delta \beta^2} \geq \|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}$ .

## I. Subspace Correction Methods for Total Variation Minimization



**Figure 3.1:** Image of size  $1920 \times 2576$  pixels which is corrupted by Gaussian blur with kernel size  $15 \times 15$  pixels and standard deviation 2, 4% salt-and-pepper noise, and Gaussian white noise with zero mean and variance 0.01. In (a) decomposition of the spatial domain into stripes and in (b) into windows.

4. If  $\alpha_T > 0$  and  $T^*T$  is symmetric positive definite with smallest Eigenvalue  $\sigma > 0$ , then the factor  $\frac{\gamma}{\beta^2}$  on the right hand side of the inequality in (3.10) is replaced by  $\alpha_T \sigma$ , cf. [6], and (3.10) reads as follows

$$\alpha_T(\sigma - \delta)\|u^* - u^{(\infty)}\|_{\ell^2(\Omega)}^2 + \alpha_T\|T(u^* - u^{(\infty)})\|_{\ell^2(\Omega)}^2 \leq \|\hat{\eta}\|_{\ell^2(\Omega)}\|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}.$$

By using once more the symmetric positive definiteness assumption on  $T^*T$  we obtain from the latter inequality that  $\alpha_T(2\sigma - \delta)\|u^* - u^{(\infty)}\|_{\ell^2(\Omega)}^2 \leq \|\hat{\eta}\|_{\ell^2(\Omega)}\|u^{(\infty)} - u^*\|_{\ell^2(\Omega)}$ . If  $2\sigma > \delta$  then we get  $\|u^* - u^{(\infty)}\|_{\ell^2(\Omega)} \leq \frac{\|\hat{\eta}\|_{\ell^2(\Omega)}}{\alpha_T(2\sigma - \delta)}$ .

□

## 3.4. Numerical Experiments

We present numerical experiments obtained by the parallel algorithm in (3.5) for the application of image deblurring, i.e.,  $S = T$  are blurring operators and  $\varphi(|\nabla u|)(\Omega) = |\nabla u|(\Omega)$  (the total variation of  $u$  in  $\Omega$ ). The minimization problems of the subdomains are implemented in the same way as described in [6] by noting that the functional to be considered in each subdomain is now the strictly convex functional in (3.3).

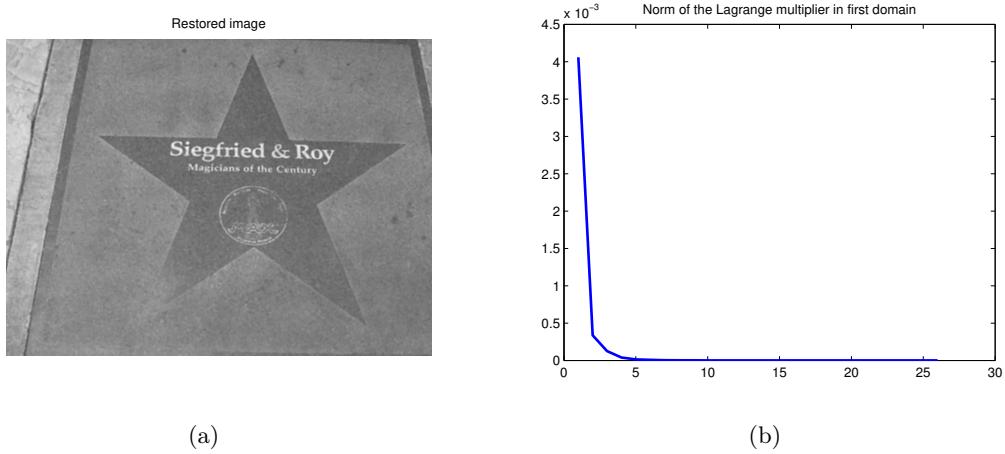
We consider an image of size  $1920 \times 2576$  pixels which is corrupted by Gaussian blur with kernel size  $15 \times 15$  pixels and standard deviation 2. Additionally 4% salt-and-pepper noise (i.e., 4% of the pixels are either flipped to black or white) and Gaussian white noise with zero mean and variance 0.01 is added.

In order to show the efficiency of the parallel algorithm in (3.5) for decomposing the spatial domain into subdomains, we compare its performance with the  $L^1$ - $L^2$ -TV algorithm presented in [6], which solves the problem on all of  $\Omega$  without any splitting. We consider splittings of the domain in stripes, cf. Figure 3.1(a), and in windows as depicted in Figure 3.1(b) for different numbers of subdomains ( $D = 4, 16, 64$ ).

### 3. Surrogate Functional Based Subspace Correction

# Domains	window-splitting	stripe-splitting
$D = 1$ ( $L^1$ - $L^2$ -TV alg.): 11944 s / 131 it		
$D = 4$ :	2374 s / 27 it	2340 s / 27 it
$D = 16$ :	2914 s / 27 it	2982 s / 27 it
$D = 64$ :	7833 s / 27 it	8797 s / 28 it

**Table 3.1:** Restoration of the image in Figure 3.1: Computational performance (CPU time in seconds and the number of iterations) for the global  $L^1$ - $L^2$ -TV algorithm and for the parallel domain decomposition algorithms with  $\alpha_1 = 0.5$ ,  $\alpha_2 = 0.4$  for different numbers of subdomains ( $D = 4, 16, 64$ ).



**Figure 3.2:** (a) Restoration of the image in Figure 3.2 by the parallel subspace correction algorithm in (3.5). (b) The progress of the minimal Lagrange multiplier  $\eta^{(n)}$ .

The algorithms are stopped as soon as the energy  $J$  reaches a significance level  $J^*$ , i.e., when  $J(u^{(n)}) \leq J^*$  for the first time. For reason of comparison we experimentally choose  $J^* = 0.059054$ , i.e., we once restored the image of interest until we observed a visually satisfying restoration and the associated energy-value as  $J^*$ . In the subspace correction algorithm as well as in the  $L^1$ - $L^2$ -TV algorithm we restore the image by setting  $\alpha_S = 0.5$ ,  $\alpha_T = 0.4$ , and  $\delta = 1.1$ . The computations are done in Matlab on a computer with 256 cores and the multithreading-option is activated.

Table 3.1 presents the computational time and number of iterations the algorithms need to fulfill the stopping criterion for different number of subdomains. We clearly see that the domain decomposition algorithm for  $D = 4, 16, 64$  subdomains is much faster than the  $L^1$ - $L^2$ -TV algorithm ( $D = 1$ ). Since a blurring operator is in general nonlocal, in each iteration  $u^{(n)}$  has been communicated to each subdomain. Therefore the communication time becomes substantial for splittings into 16 or more domains such that the algorithm needs more time to reach the stopping criterion. In Figure 3.2 we depict the progress of the minimal Lagrange multiplier  $\eta^{(n)} := \min_i \{\|\eta_i^{(n)}\|_{\ell^2(\Omega)}\}$ , which indicates that the parallel algorithm indeed converges to a minimizer of the functional  $J$ .

## I. Subspace Correction Methods for Total Variation Minimization

### 3.5. References

- [1] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. Oxford: Clarendon Press. xviii, Oxford, New York 2000.
- [2] P. L. Combettes, V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation* 4:1168–1200, 2005.
- [3] M. Fornasier, Y. Kim, A. Langer, and C.-B. Schönlieb. Wavelet decomposition method for  $L_2/TV$ -image deblurring. *SIAM Journal on Imaging Sciences*, 5(3):857–885, 2012.
- [4] M. Fornasier, A. Langer, and C.-B. Schönlieb. A convergent overlapping domain decomposition method for total variation minimization. *Numerische Mathematik*, 116(4):645–685, 2010.
- [5] M. Fornasier, and C.-B. Schönlieb. Subspace correction methods for total variation and  $\ell_1$ -minimization. *SIAM Journal on Numerical Analysis*, 47(5):3397–3428, 2009.
- [6] M. Hintermüller, and A. Langer. Subspace correction methods for non-smooth and non-additive convex variational problems in image processing. SFB-Report 2012-021, Institute for Mathematics and Scientific Computing, Karl-Franzens-University Graz, 2012.
- [7] J. B. Hiriart-Urruty, and C. Lemaréchal. *Convex Analysis and Minimization Algorithms I*, Vol. 305 of *Grundlehren der mathematischen Wissenschaften*. Springer, Berlin, 1996.
- [8] Y. Nesterov. Gradient methods for minimizing composite objective function. *CORE Discussion Paper 2007/76*, 2007.
- [9] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, NJ, 1970.

## 4. Nonoverlapping Domain Decomposition Methods for Dual Total Variation Based Image Denoising

**Bibliographic Note:** The content of this chapter is published in the article [AL3]: M. Hintermüller and A. Langer. Non-overlapping domain decomposition methods for dual total variation based image denoising. *Journal of Scientific Computing*, 62(2):456–481, 2015.

**Summary:** In this paper nonoverlapping domain decomposition methods for the pre-dual total variation minimization problem are introduced. Both parallel and sequential approaches are proposed for these methods for which convergence to a minimizer of the original problem is established. The associated subproblems are solved by a semismooth Newton method. Several numerical experiments are presented, which show the successful application of the sequential and parallel algorithm for image denoising.

### 4.1. Introduction

In the context of image denoising, Rudin, Osher and Fatemi [37] proposed to use the total variation as a regularization technique. In this approach, the recovery of an unknown image is based on solving the constrained minimization problem

$$\begin{aligned} & \min \int_{\Omega} |Du| \quad \text{over } u \in BV(\Omega) \\ & \text{subject to (s.t.) } \int_{\Omega} u dx = \int_{\Omega} f dx \quad \text{and} \quad \int_{\Omega} (u - f)^2 dx = \sigma^2, \end{aligned} \tag{4.1}$$

where  $\Omega \subset \mathbb{R}^d$ ,  $d = 2, 3$ , is a simply connected domain with Lipschitz boundary  $\partial\Omega$ ,  $f \in L^2(\Omega)$  is the observed image, and  $\sigma$  is the standard deviation of Gaussian white noise with zero mean. Moreover,  $BV(\Omega)$  denotes the *space of functions of bounded variation* [1, 18] with  $u \in BV(\Omega)$  if and only if  $u \in L^1(\Omega)$  and the total variation

$$\int_{\Omega} |Du| = \sup \left\{ \int_{\Omega} u \operatorname{div} \mathbf{v} : \mathbf{v} \in (C_c^1(\Omega))^d, \|\mathbf{v}\|_{\infty} \leq 1 \right\}$$

is finite. Here,  $C_c^1(\Omega)$  denotes the space of  $C^1$ -functions with compact support in  $\Omega$  and  $\|\mathbf{v}\|_{\infty} = \sup_x \sqrt{\sum_i v_i^2(x)}$ . The space  $BV(\Omega)$  endowed with the norm  $\|u\|_{BV(\Omega)} = \|u\|_{L^1(\Omega)} + \int_{\Omega} |Du|$  is a Banach space [24].

It is well-established that the nonsmooth and nonadditive total variation preserves edges and discontinuities in images [6, 11], and many efficient numerical methods have been proposed for solving the total variation minimization problem (4.1); see for example [2, 5, 7, 8, 9, 14, 15,

## I. Subspace Correction Methods for Total Variation Minimization

16, 29, 33, 34, 43]. A common strategy to solve (4.1) is to minimize an energy that consists of a data-fidelity term, which enforces the consistency between the recovered and the measured image, and the total variation as the regularization term, i.e., one solves

$$\min \frac{1}{2} \int_{\Omega} |u - f|^2 dx + \alpha \int_{\Omega} |Du| \quad \text{over } u \in BV(\Omega) \quad (4.2)$$

with  $\alpha > 0$ .

Due to the continuous improvement of hardware, the dimensionality of images and measurements in general is increasing, resulting in big data sets that need to be processed. While existing state-of-the-art numerical algorithms for the solution of (4.2) perform well on small- and medium-scale problems, none of these are capable of addressing extremely large problems in realistic CPU-time. In such a situation subspace correction, domain decomposition and block-coordinate descent methods are fundamental as they allow for splitting the computational workload and solving a sequence of smaller problems, possibly distributed on multiple cores. We emphasize, however, that well-known approaches as those in [4, 10] are not directly applicable to the problem in (4.2) due to the nonsmoothness and nonadditivity of the total variation regularization. The subspace correction approaches in [40, 41], which were originally introduced for smooth and convex problems, have been used in [13, 44] to solve a smoothed version of (4.2). Lately these strategies have been adapted in [17] to solve the minimization problem in (4.2) directly by graph cuts. However, the suggested domain decomposition, leading to homogeneous Dirichlet boundary conditions on the respective subdomains, is not suitable for nonoverlapping decompositions. In [12] a splitting method for minimizing the nonlocal total variation (see [23, 35, 46, 47] and the references therein for more information on nonlocal total variation) was described. Nevertheless, for the decomposition approaches in [12, 17] no rigorous theoretical analysis was carried out. In particular, the convergence proofs supplied in [40, 41] cannot be used for problems of the type (4.2) due to nonsmoothness and lack of strong convexity.

Recently in [17, 19, 20, 21, 22, 26, 27, 31, 32, 38] domain decomposition strategies were introduced for minimizing objectives including the total variation. In this context, the major difficulty lies in the correct treatment of the interfaces of the domain patches, i.e. the preservation of crossing discontinuities and the correct matching where the solution is continuous. While in these papers an implementation guaranteeing convergence and monotonic decay of the objective energy is provided, convergence to the global minimizer of the underlying problem cannot be ensured, in general. For one-dimensional problems, in [21] a proof is presented which establishes convergence of an overlapping domain decomposition algorithm to the global solution. However, it is not clear yet how to extend this proof to any finite dimensional space without introducing additional (possibly restrictive) assumptions.

In [19], for a wavelet decomposition method an additional condition is invoked which allows to establish global optimality of a limit point obtained by the decomposition method. Unfortunately, despite the good practical behavior of the method, this condition cannot be ensured to hold in general, as counterexamples have shown. Thus, with the aforementioned condition one can only check *a posteriori* whether the algorithm found the global minimizer or whether it failed to do so.

Additionally to the convergence and monotonic decay of the energy the authors derived in [26, 27] an estimate of the distance of the limit point obtained from domain decomposition methods to the true global minimizer. With the help of this estimate, it is demonstrated by numerical experiments that the sequence generated by the domain decomposition algorithm

#### 4. Domain Decomposition for Dual Total Variation

indeed approaches the global minimizer of the objective functional. However, a proof of convergence of such a domain decomposition method to the minimizer of the global problem seems elusive. Here the main difficulty lies in the presence of the nonadditive and nonsmooth total variation term.

In order to tackle the difficulties due to the minimization of a nonsmooth and nonadditive objective over a nonreflexive Banach space in (4.2), we consider the pre-dual problem of (4.2); see [25, 28] for the derivation of the latter. In fact, the pre-dual reads:

$$\begin{aligned} & \min \frac{1}{2} \|\operatorname{div} \mathbf{p} + f\|_{L^2(\Omega)}^2 \quad \text{over } \mathbf{p} \in H_0(\operatorname{div}, \Omega) \\ & \text{s.t. } -\alpha \mathbf{1} \leq \mathbf{p}(x) \leq \alpha \mathbf{1} \text{ for almost every } x \in \Omega, \end{aligned} \tag{4.3}$$

where  $H_0(\operatorname{div}, \Omega) := \{\mathbf{v} \in \mathbb{L}^2(\Omega) : \operatorname{div} \mathbf{v} \in L^2(\Omega), \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega\}$  and  $\mathbf{n}$  is the outward unit normal on  $\partial\Omega$ .

The smooth objective of  $L^2$ -type and the pointwise constraints turn out to be more amenable to domain decomposition methods than the structure of (4.2). In fact, in the present paper we introduce nonoverlapping domain decomposition methods for the problem in (4.3) and show their convergence to a global minimizer in a discrete setting. Our approach is closely related to the domain decomposition method considered in [39, 41], where the strong convexity assumption on the objective was essential to show convergence of the domain decomposition algorithm to a global minimum. We emphasize that our problem in (4.3) is convex but not strongly convex and hence the convergence theory used in [39, 41] is not directly applicable to the problem considered in this paper. In [42] a convergence theory for coordinate gradient descent methods was established for a class of minimization problems. In that paper, convergence to a global minimizer could be proven only under the assumption of strict convexity of the objective. This latter assumption, however, does not hold for the problem in (4.3). Moreover, the convergence analysis presented in [42] is not directly adaptable to our problem. Hence for our proposed domain decomposition methods a different convergence analysis is needed.

In our numerical experiments we show that our domain decomposition approach for the dual problem (4.3) converges faster than the domain decomposition approach in [26] for the primal problem (4.2). Hence the proposed splitting method is not only superior from a theoretical point of view, since we can show convergence to the global solution, but also practically. These reasons make the domain decomposition algorithm for the (pre-)dual problem applicable to 3D imaging problems, where a large number of data have to be processed.

The rest of the paper is organized as follows: In Section 4.2 we present the main notations used throughout the rest of the paper. In Section 4.3 we introduce the sequential and parallel nonoverlapping domain decomposition method and show its convergence to a minimizer of the global problem. Details on the numerical implementation of the solvers for the proposed domain decomposition methods are described in Section 4.4. Finally in Section 4.5 we show sequential and parallel numerical experiments for image denoising. We compare the computational performance of the proposed domain decomposition algorithm of the dual problem with the domain decomposition algorithm for the problem in (4.2) introduced in [26]. Moreover, we show the successful application of the proposed algorithm for 3D medical data.

## I. Subspace Correction Methods for Total Variation Minimization

### 4.2. Notations

In the rest of the paper we work in a finite dimensional setting by considering a finite regular (pixel) mesh as a discretization of a domain  $\Omega$ . For simplicity, we take  $\Omega := [0, 1]^2$ , but an extension to any domain in any dimension  $d$  is straightforward, as it is done for an example in Section 4.5.3 for  $d = 3$ . Further we approximate functions  $\mathbf{p}$  by discrete functions, again denoted by  $\mathbf{p}$ . Since we are mainly interested in imaging problems, it is sufficient to introduce our main notations for a two-dimensional space only.

In our discrete setting,  $\Omega$  is a mesh in  $\mathbb{R}^2$  of size  $N \times N$ , where  $N \in \mathbb{N}$  with mesh size  $h = x_{i,j} - x_{i+1,j} = x_{i,j} - x_{i,j+1}$  for  $x_{i,j} \in \Omega$ . The considered “function spaces” are  $\mathcal{H} = \mathbb{R}^{N \times N}$  and  $V = \mathcal{H}^2$ , with corresponding norm

$$\|u\|_{\mathcal{H}} = \|u\|_{\ell^2(\Omega)} = \left( h^2 \sum_{x \in \Omega} |u(x)|^2 \right)^{1/2}, \quad \text{for all } u \in \mathcal{H},$$

and

$$\|\mathbf{p}\|_V = \left( h^2 \sum_{x \in \Omega} |p^1(x)|^2 + |p^2(x)|^2 \right)^{1/2}, \quad \text{for all } \mathbf{p} = (p^1, p^2) \in V,$$

respectively. We define the scalar product of  $u, v \in \mathcal{H}$  and of  $\mathbf{p}, \mathbf{q} \in V$ , respectively, by

$$\langle u, v \rangle_{\mathcal{H}} = h^2 \sum_{x \in \Omega} u(x)v(x), \quad \text{and} \quad \langle \mathbf{p}, \mathbf{q} \rangle_V = h^2 \sum_{x \in \Omega} \langle \mathbf{p}(x), \mathbf{q}(x) \rangle_{\mathbb{R}^2}$$

with  $\langle y, z \rangle_{\mathbb{R}^2} = \sum_{j=1}^2 y_j z_j$  for every  $y = (y_1, y_2) \in \mathbb{R}^2$  and  $z = (z_1, z_2) \in \mathbb{R}^2$ .

The discrete gradient  $\nabla_{\Omega} u$  is denoted by  $(\nabla_{\Omega} u)(x) = ((\nabla_{\Omega} u)^1(x), (\nabla_{\Omega} u)^2(x))$  with

$$(\nabla_{\Omega} u)^1(x_{i,j}) = \frac{1}{h} \cdot \begin{cases} u(x_{i+1,j}) - u(x_{i,j}) & \text{if } i < N, \\ 0 & \text{if } i = N, \end{cases}$$

and

$$(\nabla_{\Omega} u)^2(x_{i,j}) = \frac{1}{h} \cdot \begin{cases} u(x_{i,j+1}) - u(x_{i,j}) & \text{if } j < N, \\ 0 & \text{if } j = N, \end{cases}$$

for all  $i = 1, \dots, N$  and  $j = 1, \dots, N$ . For an operator  $T$  we denote by  $T^*$  its adjoint (or transpose). Further we introduce the *discrete divergence*  $\operatorname{div}_{\Omega} : V \rightarrow \mathcal{H}$  as  $\operatorname{div}_{\Omega} := -\nabla_{\Omega}^*$  ( $\nabla_{\Omega}^*$  is the adjoint of the gradient  $\nabla_{\Omega}$ ), in analogy to the continuous setting. In our case, the discrete divergence operator is explicitly given by

$$\begin{aligned} (\operatorname{div}_{\Omega} \mathbf{p})(x_{i,j}) &= \frac{1}{h} \begin{cases} p^1(x_{i,j}) - p^1(x_{i-1,j}) & \text{if } 1 < i < N, \\ p^1(x_{i,j}) & \text{if } i = 1, \\ -p^1(x_{i-1,j}) & \text{if } i = N, \end{cases} \\ &\quad + \frac{1}{h} \begin{cases} p^2(x_{i,j}) - p^2(x_{i,j-1}) & \text{if } 1 < j < N, \\ p^2(x_{i,j}) & \text{if } j = 1, \\ -p^2(x_{i,j-1}) & \text{if } j = N, \end{cases} \end{aligned}$$

for every  $\mathbf{p} = (p^1, p^2) \in V$ . Note, that with these definitions of the gradient and divergence we have that  $\langle \mathbf{p}, \nabla_{\Omega} u \rangle_V = \langle -\operatorname{div}_{\Omega} \mathbf{p}, u \rangle_{\mathcal{H}}$ . In the sequel we also use the notations  $\nabla := \nabla_{\Omega}$  and  $\operatorname{div} := \operatorname{div}_{\Omega}$  when it is clear from the context that all of  $\Omega$  is considered.

#### 4. Domain Decomposition for Dual Total Variation

We will often use the symbols  $\mathbf{1}$  and  $\mathbf{0}$  to indicate the constant vectors with entry values 1 and 0, respectively.

For a convex functional  $\mathcal{J} : \mathcal{H} \rightarrow \bar{\mathbb{R}} := \mathbb{R} \cup \{-\infty, +\infty\}$ , we define the *subdifferential* of  $\mathcal{J}$  at  $v \in \mathcal{H}$  as the set valued mapping  $\partial\mathcal{J} : \mathcal{H} \rightarrow 2^{\mathcal{H}}$

$$\partial\mathcal{J}(v) := \begin{cases} \emptyset & \text{if } \mathcal{J}(v) = \infty, \\ \{v^* \in \mathcal{H} : \langle v^*, u - v \rangle_{\mathcal{H}} + \mathcal{J}(v) \leq \mathcal{J}(u) \quad \forall u \in \mathcal{H}\} & \text{otherwise.} \end{cases}$$

Whenever  $w \in \mathcal{H}$  is fixed,  $\partial\mathcal{J}(\cdot + w)(v)$ , for  $v \in \mathcal{H}$ , denotes the evaluation of  $\partial\mathcal{J}$  at  $v + w$ . It is clear from this definition that  $0 \in \partial\mathcal{J}(v)$  if and only if  $v$  is a minimizer of  $\mathcal{J}$ . Since we have to handle different spaces, namely  $\mathcal{H}, V$ , it is sometimes useful to distinguish the spaces in the definition of the subdifferential by imposing a subscript  $\partial_{\mathcal{H}}\mathcal{J}$  for the subdifferential considered on the space  $\mathcal{H}$ , for instance.

### 4.3. Nonoverlapping Domain Decomposition of the (Pre-)Dual Problem

We consider the splitting of  $\Omega$  into two disjoint subdomains  $\Omega_1$  and  $\Omega_2$  such that  $\Omega_2 = \Omega \setminus \Omega_1$  and  $\Omega = \Omega_1 \cup \Omega_2$ . For consistency with the definitions of the gradient and divergence operator, we assume that the subdomains  $\Omega_i$ ,  $i = 1, 2$ , as well as  $\Omega$  are discrete rectangles. We stress that the shape of the domains as well as the decomposition into two subdomains represents by no means a restriction, but rather simplifies the presentation. Associated with this decomposition we define  $V_i := \{\mathbf{v} \in V : \mathbf{v}(x) = \mathbf{0} \text{ for } x \in \Omega \setminus \Omega_i\}$  such that  $V = V_1 \oplus V_2$  and by  $\pi_{V_i}$  we denote the corresponding orthogonal projection onto  $V_i$ . With this splitting we aim to solve the discrete problem

$$\begin{aligned} & \min \frac{1}{2} \|\operatorname{div} \mathbf{p} + f\|_{\mathcal{H}}^2 \quad \text{over } \mathbf{p} \in V \\ & \text{s.t. } -\alpha \mathbf{1} \leq \mathbf{p}(x) \leq \alpha \mathbf{1} \text{ for all } x \in \Omega, \end{aligned} \tag{4.4}$$

by the following iterative scheme:

**Algorithm 1.**

- *Initialize:*  $\mathbf{p}^{(0)} =: \mathbf{p}_1^{(0)} + \mathbf{p}_2^{(0)} \in V_1 \oplus V_2$  with  $-\alpha \mathbf{1} \leq \mathbf{p}^{(0)}(x) \leq \alpha \mathbf{1}$  for all  $x \in \Omega$ .
- *Iterate:* For  $k = 0, 1, \dots$  do

$$\left\{ \begin{array}{l} \mathbf{p}_1^{(k+1)} \in \arg \min_{\mathbf{p}_1 \in V_1} \left\{ \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1 + \mathbf{p}_2^{(k)}) + f\|_{\mathcal{H}}^2 : -\alpha \mathbf{1} \leq \mathbf{p}_1(x) + \mathbf{p}_2^{(k)}(x) \leq \alpha \mathbf{1} \right\}, \\ \mathbf{p}_2^{(k+1)} \in \arg \min_{\mathbf{p}_2 \in V_2} \left\{ \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1^{(k+1)} + \mathbf{p}_2) + f\|_{\mathcal{H}}^2 : -\alpha \mathbf{1} \leq \mathbf{p}_1^{(k+1)}(x) + \mathbf{p}_2(x) \leq \alpha \mathbf{1} \right\}, \\ \mathbf{p}^{(k+1)} := \mathbf{p}_1^{(k+1)} + \mathbf{p}_2^{(k+1)}. \end{array} \right.$$

For the ease of notation, from now on we use  $-\alpha \mathbf{1} \leq \mathbf{p}(x) \leq \alpha \mathbf{1}$  instead of  $-\alpha \mathbf{1} \leq \mathbf{p}(x) \leq \alpha \mathbf{1}$  for all  $x \in \Omega$ .

**Remark 4.1.** According to the notations in [10, 45] we call Algorithm 1 a nonoverlapping domain decomposition algorithm, as it is based on a partitioning of the domain  $\Omega$  into nonoverlapping subdomains. Note that for solving the associated optimization problems, information outside of the respective subdomain is needed. More precisely, the minimization problems in

## I. Subspace Correction Methods for Total Variation Minimization

*Algorithm 1 need to be solved at least on the associated subdomain plus a small stripe around the interfaces, see Section 4.4. However, the obtained solution of the respective subdomain problem will be in  $V_i$  for  $i = 1, 2$  and hence it has only support in  $\Omega_i$ .*

Several basic properties of Algorithm 1 are summarized next.

**Proposition 4.2.** *Let  $J(\mathbf{p}) := \frac{1}{2} \|\operatorname{div} \mathbf{p} + f\|_{\mathcal{H}}^2$  and  $\mathbf{p}^{(k+\frac{1}{2})} := \mathbf{p}_1^{(k+1)} + \mathbf{p}_2^{(k)}$ . Then Algorithm 1 generates sequences  $(\mathbf{p}^{(k)})_k \subset V$  and  $(\mathbf{p}_i^{(k)})_k \subset V_i$  for  $i = 1, 2$  with the following properties:*

- (i)  $J(\mathbf{p}^{(k)}) \geq J(\mathbf{p}^{(k+\frac{1}{2})}) \geq J(\mathbf{p}^{(k+1)})$  for all  $k \in \mathbb{N}$ ;
- (ii) The sequences  $(J(\mathbf{p}^{(k)}))_k$  and  $(J(\mathbf{p}^{(k+\frac{1}{2})}))_k$  converge;
- (iii)  $\lim_{k \rightarrow \infty} (J(\mathbf{p}^{(k+\frac{1}{2})}) - J(\mathbf{p}^{(k)})) = 0$ ;
- (iv) The sequences  $(\mathbf{p}^{(k)})_k$  and  $(\mathbf{p}^{(k+\frac{1}{2})})_k$  are bounded in  $V$  and hence have subsequences that converge in  $V$ ;
- (v) The sequences  $(\mathbf{p}_i^{(k)})_k$  for  $i = 1, 2$  are bounded in  $V$ .

*Proof.* (i) The assertion follows from the optimality properties of  $\mathbf{p}_1^{(k)}$  and  $\mathbf{p}_2^{(k)}$ .

- (ii)+(iii) The statements follow directly from (i) and the fact that  $J(\cdot)$  is bounded from below.
- (iv) Since  $-\alpha \mathbf{1} \leq \mathbf{p}_1^{(k)}(x) + \mathbf{p}_2^{(k)}(x) \leq \alpha \mathbf{1}$  and  $-\alpha \mathbf{1} \leq \mathbf{p}_1^{(k+1)}(x) + \mathbf{p}_2^{(k)}(x) \leq \alpha \mathbf{1}$  for all  $x \in \Omega$  it follows that

$$\|\mathbf{p}^{(k)}\|_V \leq \alpha \sqrt{2} \text{ and } \|\mathbf{p}^{(k+\frac{1}{2})}\|_V \leq \alpha \sqrt{2} \quad \text{for all } k \in \mathbb{N}.$$

- (v) Since  $\mathbf{p}_i^{(k)} \in V_i$  it follows from the boundedness of  $(\mathbf{p}^{(k)})_k$ , see (iv), that

$$2\alpha^2 \geq \|\mathbf{p}^{(k)}\|_V^2 = \|\mathbf{p}_1^{(k)} + \mathbf{p}_2^{(k)}\|_V^2 = \|\mathbf{p}_1^{(k)}\|_V^2 + \|\mathbf{p}_2^{(k)}\|_V^2$$

and hence  $(\mathbf{p}_i^{(k)})_k$  is bounded in  $V$  for  $i = 1, 2$ . □

**Proposition 4.3.** *For  $i = 1, 2$  let  $(\mathbf{p}_i^{(k)})_k$  be generated by Algorithm 1 and  $(\mathbf{p}_i^{(k_l)})_l$  be a convergent subsequence with limit point  $\mathbf{p}_i^{(\infty,0)}$ . Then there exists an accumulation point  $\mathbf{p}_i^{(\infty,1)}$  of the sequence  $(\mathbf{p}_i^{(k_l+1)})_l$  such that*

- (i)  $\mathbf{p}_1^{(\infty,0)}$  and  $\mathbf{p}_1^{(\infty,1)}$  are minimizers of

$$\begin{aligned} & \min \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1 + \mathbf{p}_2^{(\infty,0)}) + f\|_{\mathcal{H}}^2 \quad \text{over } \mathbf{p}_1 \in V_1 \\ & \text{s.t. } -\alpha \mathbf{1} \leq \mathbf{p}_1(x) + \mathbf{p}_2^{(\infty,0)}(x) \leq \alpha \mathbf{1}, \end{aligned}$$

and

#### 4. Domain Decomposition for Dual Total Variation

(ii)  $\mathbf{p}_2^{(\infty,0)}$  and  $\mathbf{p}_2^{(\infty,1)}$  are minimizers of

$$\begin{aligned} & \min \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2) + f\|_{\mathcal{H}}^2 \quad \text{over } \mathbf{p}_2 \in V_2 \\ & \text{s.t. } -\alpha \mathbf{1} \leq \mathbf{p}_1(x)^{(\infty,1)} + \mathbf{p}_2(x) \leq \alpha \mathbf{1}, \end{aligned}$$

*Proof.* Set  $F(\mathbf{p}) := \frac{1}{2} \|\operatorname{div}(\mathbf{p}) + f\|_{\mathcal{H}}^2 + I_{[-\alpha, \alpha]}(\mathbf{p})$ , where

$$I_{[-\alpha, \alpha]}(\mathbf{p}) = \begin{cases} 0 & \text{if } -\alpha \mathbf{1} \leq \mathbf{p}(x) \leq \alpha \mathbf{1} \text{ for all } x \in \Omega, \\ \infty & \text{otherwise.} \end{cases} \quad (4.5)$$

Then the minimization problem in  $V_1$  is equivalent to

$$\min F(\mathbf{p}_1 + \mathbf{p}_2^{(k)}) \quad \text{over } \mathbf{p}_1 \in V_1.$$

Note that  $F$  is convex, lower semicontinuous, bounded from below, and monotonically decreasing with respect to the iterations, i.e.,  $F(\mathbf{p}^{(k)}) \geq F(\mathbf{p}^{(k+1)})$  for all  $k \in \mathbb{N}$ .

(i) Since  $(\mathbf{p}_2^{(k)})_k$  is bounded, there exists a convergent subsequence  $(\mathbf{p}_2^{(k_l)})_l$  with limit  $\mathbf{p}_2^{(\infty,0)}$ . Further  $(\mathbf{p}_1^{(k_l+1)})_l$  is bounded and hence there exists a convergent subsequence  $(\mathbf{p}_1^{(k_{l_j}+1)})_j$  with limit  $\mathbf{p}_1^{(\infty,1)}$ .

By the minimality property of  $\mathbf{p}_1^{(k_{l_j}+1)}$  we have that  $0 \in \partial_{V_1} F(\cdot + \mathbf{p}_2^{(k_{l_j})})(\mathbf{p}_1^{(k_{l_j}+1)})$ . Then by [36, Theorem 24.4, p. 233] we obtain that  $0 \in \partial_{V_1} F(\cdot + \mathbf{p}_2^{(\infty,0)})(\mathbf{p}_1^{(\infty,1)})$  and hence by convexity  $\mathbf{p}_1^{(\infty,1)} \in \arg \min_{\mathbf{p}_1 \in V_1} F(\mathbf{p}_1 + \mathbf{p}_2^{(\infty,0)})$ .

Analogously we get that  $0 \in \partial_{V_2} F(\mathbf{p}_1^{(\infty,1)} + \cdot)(\mathbf{p}_2^{(\infty,1)})$ , i.e.,  $\mathbf{p}_2^{(\infty,1)} \in \arg \min_{\mathbf{p}_2 \in V_2} F(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2)$ .

Moreover, by the monotone decrease of the energy we have that

$$F(\mathbf{p}^{(k)}) \geq F(\mathbf{p}_1^{(k+1)} + \mathbf{p}_2^{(k)}) \geq F(\mathbf{p}^{(k+1)}) \quad \text{for all } k \in \mathbb{N}$$

and hence

$$F(\mathbf{p}^{(k)}) - F(\mathbf{p}^{(k+1)}) \geq F(\mathbf{p}_1^{(k+1)} + \mathbf{p}_2^{(k)}) - F(\mathbf{p}_1^{(k+1)} + \mathbf{p}_2^{(k+1)}) \geq 0.$$

Replacing  $(k)$  and  $(k+1)$  by  $(k_{l_j})$  and  $(k_{l_j}+1)$ , respectively, we get

$$0 \leq F(\mathbf{p}_1^{(k_{l_j}+1)} + \mathbf{p}_2^{(k_{l_j})}) - F(\mathbf{p}_1^{(k_{l_j}+1)} + \mathbf{p}_2^{(k_{l_j}+1)}) \leq F(\mathbf{p}_1^{(k_{l_j}+1)} + \mathbf{p}_2^{(k_{l_j})}) - F(\mathbf{p}_1^{(k_{l_j+1})} + \mathbf{p}_2^{(k_{l_j+1})}).$$

Since  $(\mathbf{p}_1^{(k_{l_j+1})})_j$  is bounded, there exists a subsequence  $(\mathbf{p}_1^{(k_{l_{j_m+1}})})_m$  with limit  $\mathbf{p}_1^{(\infty,0)}$  and we have

$$F(\mathbf{p}_1^{(k_{l_{j_m}}+1)} + \mathbf{p}_2^{(k_{l_{j_m}})}) - F(\mathbf{p}_1^{(k_{l_{j_m+1}})} + \mathbf{p}_2^{(k_{l_{j_m+1}})}) \geq 0.$$

## I. Subspace Correction Methods for Total Variation Minimization

Proposition 4.2(ii) implies  $\lim_{k \rightarrow \infty} [F(\mathbf{p}^{(k)}) - F(\mathbf{p}^{(k+r)})] = 0$  for all  $r \in \mathbb{N}$ . Consequently

$$\begin{aligned} 0 &\leq \lim_{m \rightarrow \infty} \left[ F(\mathbf{p}_1^{(k_{l_{j_m}}+1)} + \mathbf{p}_2^{(k_{l_{j_m}})}) - F(\mathbf{p}_1^{(k_{l_{j_{m+1}}})} + \mathbf{p}_2^{(k_{l_{j_{m+1}}})}) \right] \\ &\leq \lim_{m \rightarrow \infty} \left[ F(\mathbf{p}_1^{(k_{l_{j_m}})} + \mathbf{p}_2^{(k_{l_{j_m}})}) - F(\mathbf{p}_1^{(k_{l_{j_{m+1}}})} + \mathbf{p}_2^{(k_{l_{j_{m+1}}})}) \right] = 0. \end{aligned}$$

Further

$$\begin{aligned} \lim_{m \rightarrow \infty} &\left[ F(\mathbf{p}_1^{(k_{l_{j_m}}+1)} + \mathbf{p}_2^{(k_{l_{j_m}})}) - F(\mathbf{p}_1^{(k_{l_{j_{m+1}}})} + \mathbf{p}_2^{(k_{l_{j_{m+1}}})}) \right] \\ &= F(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) - F(\mathbf{p}_1^{(\infty,0)} + \mathbf{p}_2^{(\infty,0)}). \end{aligned}$$

This implies

$$F(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) - F(\mathbf{p}_1^{(\infty,0)} + \mathbf{p}_2^{(\infty,0)}) = 0.$$

Since  $\mathbf{p}_1^{(\infty,1)}$  is a solution of

$$\begin{aligned} \min \frac{1}{2} &\| \operatorname{div}(\mathbf{p}_1 + \mathbf{p}_2^{(\infty,0)}) + f \|_{\mathcal{H}}^2 \quad \text{over } \mathbf{p}_1 \in V_1 \\ \text{s.t. } &-\alpha \mathbf{1} \leq \mathbf{p}_1(x) + \mathbf{p}_2^{(\infty,0)}(x) \leq \alpha \mathbf{1}, \end{aligned}$$

so is  $\mathbf{p}_1^{(\infty,0)}$ .

(ii) By similar arguments one proves the second assertion. □

Next we show the convergence of Algorithm 1 to a minimizer of the (pre-)dual problem (4.4).

**Theorem 4.4.** *The sequence  $(\mathbf{p}^{(k)})_k$  generated by Algorithm 1 has an accumulation point  $\mathbf{p}^{(\infty)} \in V$  which is a solution of the problem in (4.4).*

*Proof.* From the optimality of  $\mathbf{p}_1^{(k+1)}$  and  $\mathbf{p}_2^{(k+1)}$  we get

$$\begin{aligned} 0 &\in -\pi_{V_1} \nabla (\operatorname{div}(\mathbf{p}_1^{(k+1)} + \mathbf{p}_2^{(k)}) + f) + \partial_{V_1} I_{[-\alpha, \alpha]}(\cdot + \mathbf{p}_2^{(k)})(\mathbf{p}_1^{(k+1)}) \\ 0 &\in -\pi_{V_2} \nabla (\operatorname{div}(\mathbf{p}_1^{(k+1)} + \mathbf{p}_2^{(k+1)}) + f) + \partial_{V_2} I_{[-\alpha, \alpha]}(\mathbf{p}_1^{(k+1)} + \cdot)(\mathbf{p}_2^{(k+1)}), \end{aligned} \tag{4.6}$$

where  $I_{[-\alpha, \alpha]}(\cdot)$  is defined as in (4.5). By similar arguments as in Proposition 4.3 we extract suitable subsequences and due to their boundedness (Proposition 4.2), the hierarchy of sequences  $(\mathbf{p}_2^{(k_l)})_l$ ,  $(\mathbf{p}_1^{(k_{l_j}+1)})_j$ ,  $(\mathbf{p}_2^{(k_{l_{j_m}}+1)})_m$  converges to limit points  $\mathbf{p}_2^{(\infty,0)}$ ,  $\mathbf{p}_1^{(\infty,1)}$ ,  $\mathbf{p}_2^{(\infty,1)}$ , respectively. Note that the divergence of a bounded sequence is again bounded, i.e., the boundedness of  $(\mathbf{p}^{(k)})_k \subset V$  implies the boundedness of  $(\operatorname{div} \mathbf{p}^{(k)})_k$ . Hence, we may replace  $k$  by  $k_{l_{j_m}}$  in (4.6). By taking the limit  $m \rightarrow \infty$  and using [36, Thm. 24.4, p. 233] we obtain

$$\begin{aligned} 0 &\in -\pi_{V_1} \nabla (\operatorname{div}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) + f) + \partial_{V_1} I_{[-\alpha, \alpha]}(\cdot + \mathbf{p}_2^{(\infty,0)})(\mathbf{p}_1^{(\infty,1)}) \\ 0 &\in -\pi_{V_2} \nabla (\operatorname{div}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,1)}) + f) + \partial_{V_2} I_{[-\alpha, \alpha]}(\mathbf{p}_1^{(\infty,1)} + \cdot)(\mathbf{p}_2^{(\infty,1)}). \end{aligned}$$

#### 4. Domain Decomposition for Dual Total Variation

By Proposition 4.3 we can replace  $\mathbf{p}_2^{(\infty,1)}$  by  $\mathbf{p}_2^{(\infty,0)}$  in the last inclusion. Let us define

$$\begin{aligned}\xi_1 &:= \pi_{V_1} \nabla(\operatorname{div}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) + f) \in \partial_{V_1} I_{[-\alpha,\alpha]}(\cdot + \mathbf{p}_2^{(\infty,0)})(\mathbf{p}_1^{(\infty,1)}) \\ \xi_2 &:= \pi_{V_2} \nabla(\operatorname{div}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) + f) \in \partial_{V_2} I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \cdot)(\mathbf{p}_2^{(\infty,0)}).\end{aligned}$$

Note that  $\xi := \xi_1 + \xi_2 = \nabla(\operatorname{div}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) + f)$ . By the definition of the subdifferential we get

$$\begin{aligned}\langle \xi_1, \mathbf{y}_1 - \mathbf{p}_1^{(\infty,1)} \rangle_V + I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) &\leq I_{[-\alpha,\alpha]}(\mathbf{y}_1 + \mathbf{p}_2^{(\infty,0)}) \quad \text{for all } \mathbf{y}_1 \in V_1, \\ \langle \xi_2, \mathbf{y}_2 - \mathbf{p}_2^{(\infty,0)} \rangle_V + I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) &\leq I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \mathbf{y}_2) \quad \text{for all } \mathbf{y}_2 \in V_2.\end{aligned}$$

Summing up the last two inequalities yields

$$\begin{aligned}\langle \xi, \mathbf{y}_1 + \mathbf{y}_2 - (\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) \rangle_V + I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) \\ \leq I_{[-\alpha,\alpha]}(\mathbf{y}_1 + \mathbf{p}_2^{(\infty,0)}) + I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \mathbf{y}_2) - I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) \quad \text{for all } \mathbf{y}_i \in V_i.\end{aligned}$$

Since  $\Omega_1$  and  $\Omega_2$  are disjoint we have that

$$I_{[-\alpha,\alpha]}(\mathbf{p}_1 + \mathbf{p}_2) = I_{[-\alpha,\alpha]}(\mathbf{p}_1) + I_{[-\alpha,\alpha]}(\mathbf{p}_2) \quad (4.7)$$

for all  $\mathbf{p}_i \in V_i$ ,  $i = 1, 2$ . By using this property we deduce

$$I_{[-\alpha,\alpha]}(\mathbf{y}_1 + \mathbf{p}_2^{(\infty,0)}) + I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \mathbf{y}_2) - I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) = I_{[-\alpha,\alpha]}(\mathbf{y}_1 + \mathbf{y}_2)$$

for all  $\mathbf{y}_i \in V_i$ ,  $i = 1, 2$ . Hence, for  $\mathbf{y} := \mathbf{y}_1 + \mathbf{y}_2$ , we have

$$\langle \xi, \mathbf{y} - (\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) \rangle_V + I_{[-\alpha,\alpha]}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) \leq I_{[-\alpha,\alpha]}(\mathbf{y}) \quad \text{for all } \mathbf{y} \in V.$$

and consequently

$$0 \in -\nabla(\operatorname{div}(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}) + f) + \partial_V I_{[-\alpha,\alpha]}(\cdot)(\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}).$$

On the other hand,  $\mathbf{p}^* \in V$  is a solution of the global problem (4.4) if and only if

$$0 \in -\nabla(\operatorname{div} \mathbf{p}^* + f) + \partial_V I_{[-\alpha,\alpha]}(\cdot)(\mathbf{p}^*).$$

Hence  $\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,0)}$  is a minimizer of the global problem. By Proposition 4.2(iii) and since  $0 \leq J(\mathbf{p}_1^{(k_{l,jm}+1)} + \mathbf{p}_2^{(k_{l,jm})}) - J(\mathbf{p}_1^{(k_{l,jm}+1)} + \mathbf{p}_2^{(k_{l,jm}+1)})$  it follows that  $\mathbf{p}_1^{(\infty,1)} + \mathbf{p}_2^{(\infty,1)}$  is also a minimizer of (4.4).  $\square$

**Remark 4.5.** (a) In the above proof we rely on the additivity of the function  $I_{[-\alpha,\alpha]}(\cdot)$  with respect to a disjoint decomposition. Actually, Theorem 4.4 can be extended to more general problems of the type

$$\frac{1}{2} \|T(\mathbf{p}) + g\|_{\mathcal{H}}^2 + P(\mathbf{p}), \quad (4.8)$$

where  $g \in \mathcal{H}$  is a given datum,  $P : V \rightarrow \mathbb{R}$  a convex, lower semicontinuous function with the splitting property (4.7) and  $T : V \rightarrow \mathcal{H}$  is a bounded linear operator.

## I. Subspace Correction Methods for Total Variation Minimization

Due to this property, our method can also handle deconvolution-type problems, i.e., (4.2) is replaced by

$$\min \frac{1}{2} \int_{\Omega} |Ku - f|^2 dx + \alpha \int_{\Omega} |Du| + \frac{\kappa}{2} \int_{\Omega} |u|^2 dx \quad \text{over } u \in BV(\Omega)$$

where  $K$  is a continuous linear operator from  $L^2(\Omega)$  to  $L^2(\Omega)$  modelling, e.g., convolution and  $\kappa \geq 0$ . We note that  $\kappa > 0$  is useful, whenever  $K^*K$  is not invertible. The (pre-)dual in the discrete setting becomes

$$\begin{aligned} & \min \frac{1}{2} \|\operatorname{div} \mathbf{p} + K^* f\|_M^2 \quad \text{over } \mathbf{p} \in V \\ & \text{s.t. } -\alpha \mathbf{1} \leq \mathbf{p}(x) \leq \alpha \mathbf{1}, \end{aligned}$$

where  $\|w\|_M^2 = w^T M w$  for  $M = (\kappa I + K^* K)^{-1}$ . Here, in a slight misuse of notation, we denote the discrete convolution operator also by  $K$ . Considering  $T := M^{\frac{1}{2}} \operatorname{div}$  and  $g := M^{\frac{1}{2}} K^* f$  in (4.8), the deconvolution problem fits into our framework.

- (b) As the considered problem (4.4) is convex but not strictly convex, our convergence analysis differs from the one in [42], where the strict convexity assumption is essential.

### 4.3.1. A Parallel Version and Its Convergence

The parallel version of Algorithm 1 reads as follows:

#### Algorithm 2.

- Initialize:  $\mathbf{p}^{(0)} =: \tilde{\mathbf{p}}_1^{(0)} + \tilde{\mathbf{p}}_2^{(0)} \in V_1 \oplus V_2$  with  $-\alpha \mathbf{1} \leq \mathbf{p}^{(0)}(x) \leq \alpha \mathbf{1}$  for all  $x \in \Omega$ ;
- Iterate: For  $k = 0, 1, \dots$  do

$$\left\{ \begin{array}{l} \mathbf{p}_1^{(k+1)} \in \arg \min_{\mathbf{p}_1 \in V_1} \left\{ \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1 + \tilde{\mathbf{p}}_2^{(k)}) + f\|_{\mathcal{H}}^2 : -\alpha \mathbf{1} \leq \mathbf{p}_1(x) + \tilde{\mathbf{p}}_2^{(k)}(x) \leq \alpha \mathbf{1} \right\}, \\ \mathbf{p}_2^{(k+1)} \in \arg \min_{\mathbf{p}_2 \in V_2} \left\{ \frac{1}{2} \|\operatorname{div}(\tilde{\mathbf{p}}_1^{(k)} + \mathbf{p}_2) + f\|_{\mathcal{H}}^2 : -\alpha \mathbf{1} \leq \mathbf{p}_2(x) + \tilde{\mathbf{p}}_1^{(k)}(x) \leq \alpha \mathbf{1} \right\}, \\ \mathbf{p}^{(k+1)} = \frac{\mathbf{p}_1^{(k+1)} + \mathbf{p}_2^{(k+1)} + \mathbf{p}^{(k)}}{2}, \\ \tilde{\mathbf{p}}_1^{(k+1)} := \pi_{V_1} \mathbf{p}^{(k+1)}, \\ \tilde{\mathbf{p}}_2^{(k+1)} := \pi_{V_2} \mathbf{p}^{(k+1)}. \end{array} \right.$$

We state similar convergence results as for the sequential algorithm and provide their proofs for the sake of completeness.

**Proposition 4.6.** Let  $J(\mathbf{p}) := \frac{1}{2} \|\operatorname{div} \mathbf{p} + f\|_{\mathcal{H}}^2$ . Then Algorithm 2 generates sequences  $(\mathbf{p}^{(k)})_k \subset V$ ,  $(\mathbf{p}_i^{(k)})_k \subset V_i$  and  $(\tilde{\mathbf{p}}_i^{(k)})_k \subset V_i$  with the following properties:

- (i)  $J(\mathbf{p}^{(k)}) \geq J(\mathbf{p}^{(k+1)})$  for all  $k \in \mathbb{N}$ ;
- (ii) The sequence  $(J(\mathbf{p}^{(k)}))_k$  converges;
- (iii) The sequence  $(\mathbf{p}^{(k)})_k$  is bounded in  $V$  and hence has a subsequence that converges in  $V$ ;
- (iv) The sequences  $(\mathbf{p}_i^{(k)})_k$  and  $(\tilde{\mathbf{p}}_i^{(k)})_k$  for  $i = 1, 2$  are bounded in  $V$ .

#### 4. Domain Decomposition for Dual Total Variation

*Proof.* (i) From the optimality properties of  $\mathbf{p}_1^{(k)}$  and  $\mathbf{p}_2^{(k)}$  we have

$$J(\mathbf{p}^{(k)}) \geq J(\mathbf{p}_1^{(k+1)} + \tilde{\mathbf{p}}_2^{(k)}) \quad \text{and} \quad J(\mathbf{p}^{(k)}) \geq J(\tilde{\mathbf{p}}_1^{(k)} + \mathbf{p}_2^{(k+1)}).$$

Adding these two inequalities and using the convexity of  $J$  yield

$$\begin{aligned} 2J(\mathbf{p}^{(k)}) &\geq \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1^{(k+1)} + \tilde{\mathbf{p}}_2^{(k)}) + f\|_{\mathcal{H}}^2 + \frac{1}{2} \|\operatorname{div}(\tilde{\mathbf{p}}_1^{(k)} + \mathbf{p}_2^{(k+1)}) + f\|_{\mathcal{H}}^2 \\ &\geq \|\operatorname{div}\left(\frac{\mathbf{p}_1^{(k+1)} + \tilde{\mathbf{p}}_2^{(k)} + \tilde{\mathbf{p}}_1^{(k)} + \mathbf{p}_2^{(k+1)}}{2}\right) + f\|_{\mathcal{H}}^2. \end{aligned}$$

The assertion follows since  $\tilde{\mathbf{p}}_2^{(k)} + \tilde{\mathbf{p}}_1^{(k)} = \mathbf{p}^{(k)}$  and by the definition of  $\mathbf{p}^{(k+1)}$  in Algorithm 2.

(ii) The statement follows directly from (i) and the fact that  $(J(\mathbf{p}^{(k)}))_k$  is bounded from below.

(iii) Since  $-\alpha\mathbf{1} \leq \mathbf{p}_1^{(k+1)}(x) + \tilde{\mathbf{p}}_2^{(k)}(x) \leq \alpha\mathbf{1}$  and  $-\alpha\mathbf{1} \leq \tilde{\mathbf{p}}_1^{(k)}(x) + \mathbf{p}_2^{(k+1)}(x) \leq \alpha\mathbf{1}$  we get that

$$-\alpha\mathbf{1} \leq \frac{\mathbf{p}_1^{(k+1)}(x) + \mathbf{p}_2^{(k+1)}(x) + \mathbf{p}^{(k)}(x)}{2} \leq \alpha\mathbf{1}$$

for all  $x \in \Omega$ . From the definition of  $\mathbf{p}^{(k+1)}$ , see Algorithm 2, we deduce  $-\alpha\mathbf{1} \leq \mathbf{p}^{(k+1)}(x) \leq \alpha\mathbf{1}$  for all  $x \in \Omega$  and hence  $\|\mathbf{p}^{(k)}\|_V \leq \alpha\sqrt{2}$ .

(iv) Since  $\tilde{\mathbf{p}}_i^{(k)}, \mathbf{p}_i^{(k)} \in V_i$  it follows from the boundedness of  $(\mathbf{p}^{(k)})_k$  and the inequality  $(a+b)^{\frac{1}{2}} \geq \frac{1}{\sqrt{2}}(a^{\frac{1}{2}} + b^{\frac{1}{2}})$  for  $a, b \geq 0$ , that

$$\alpha\sqrt{2} \geq \|\mathbf{p}^{(k)}\|_V = \|\tilde{\mathbf{p}}_1^{(k)} + \tilde{\mathbf{p}}_2^{(k)}\|_V \geq \frac{1}{\sqrt{2}}(\|\tilde{\mathbf{p}}_1^{(k)}\|_V + \|\tilde{\mathbf{p}}_2^{(k)}\|_V)$$

and

$$\begin{aligned} 2\alpha\sqrt{2} &\geq 2\|\mathbf{p}^{(k)}\|_V = \|\mathbf{p}_1^{(k)} + \mathbf{p}_2^{(k)} + \mathbf{p}^{(k-1)}\|_V \geq \|\mathbf{p}_1^{(k)} + \mathbf{p}_2^{(k)}\|_V - \|\mathbf{p}^{(k-1)}\|_V \\ &\geq \frac{1}{\sqrt{2}}(\|\mathbf{p}_1^{(k)}\|_V + \|\mathbf{p}_2^{(k)}\|_V) - \|\mathbf{p}^{(k-1)}\|_V \end{aligned}$$

and hence  $(\tilde{\mathbf{p}}_i^{(k)})_k$  and  $(\mathbf{p}_i^{(k)})_k$  are bounded in  $V$  for  $i = 1, 2$ .

□

**Proposition 4.7.** For  $i = 1, 2$  let  $(\mathbf{p}_i^{(k)})_k$  be generated by Algorithm 2 and  $(\mathbf{p}_i^{(k_l)})_l$  be a convergent subsequence with limit point  $\mathbf{p}_i^{(\infty, 0)}$ . Then there exists an accumulation point  $\mathbf{p}_i^{(\infty, 1)}$  of the sequence  $(\mathbf{p}_i^{(k_l+1)})_l$  such that

(i)  $\tilde{\mathbf{p}}_1^{(\infty, 0)}$  and  $\mathbf{p}_1^{(\infty, 1)}$  are minimizers of

$$\begin{aligned} &\min \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1 + \tilde{\mathbf{p}}_2^{(\infty, 0)}) + f\|_{\mathcal{H}}^2 \quad \text{over } \mathbf{p}_1 \in V_1 \\ &\text{s.t. } -\alpha\mathbf{1} \leq \mathbf{p}_1(x) + \tilde{\mathbf{p}}_2^{(\infty, 0)}(x) \leq \alpha\mathbf{1}, \end{aligned}$$

and

## I. Subspace Correction Methods for Total Variation Minimization

(ii)  $\tilde{\mathbf{p}}_2^{(\infty,0)}$  and  $\mathbf{p}_2^{(\infty,1)}$  are minimizers of

$$\begin{aligned} & \min \frac{1}{2} \|\operatorname{div}(\tilde{\mathbf{p}}_1^{(\infty,0)} + \mathbf{p}_2) + f\|_{\mathcal{H}}^2 \quad \text{over } \mathbf{p}_2 \in V_2 \\ & \text{s.t. } -\alpha \mathbf{1} \leq \mathbf{p}_2(x) + \tilde{\mathbf{p}}_1^{(\infty,0)}(x) \leq \alpha \mathbf{1}, \end{aligned}$$

*Proof.* Set  $F(\mathbf{p}) := \frac{1}{2} \|\operatorname{div}(\mathbf{p}) + f\|_{\mathcal{H}}^2 + I_{[-\alpha, \alpha]}(\mathbf{p})$  with  $I_{[-\alpha, \alpha]}$  defined as in (4.5). Then the minimization problem in  $V_1$  is equivalent to  $\min_{\mathbf{p}_1 \in V_1} F(\mathbf{p}_1 + \tilde{\mathbf{p}}_2^{(k)})$ . Note that  $F$  is convex, lower semicontinuous, bounded from below, and monotonically decreasing with respect to the iterations, i.e.,  $F(\mathbf{p}^{(k)}) \geq F(\mathbf{p}^{(k+1)})$  for all  $k \in \mathbb{N}$ .

Since  $(\tilde{\mathbf{p}}_2^{(k)})_k$  is bounded, there exists a convergent subsequence  $(\tilde{\mathbf{p}}_2^{(k_l)})_l$  with limit  $\tilde{\mathbf{p}}_2^{(\infty,0)}$ . Further  $(\mathbf{p}_1^{(k_l+1)})_l$  is bounded and hence there exists a convergent subsequence  $(\mathbf{p}_1^{(k_l+1)})_j$  with limit  $\mathbf{p}_1^{(\infty,1)}$ .

By the minimality property of  $\mathbf{p}_1^{(k_l+1)}$  we have that  $0 \in \partial_{V_1} F(\cdot + \tilde{\mathbf{p}}_2^{(k_l)}) (\mathbf{p}_1^{(k_l+1)})$ . Then by [36, Theorem 24.4, p. 233] we obtain that  $0 \in \partial_{V_1} F(\cdot + \tilde{\mathbf{p}}_2^{(\infty,0)}) (\mathbf{p}_1^{(\infty,1)})$  and hence by convexity  $\mathbf{p}_1^{(\infty,1)} \in \arg \min_{\mathbf{p}_1 \in V_1} F(\mathbf{p}_1 + \tilde{\mathbf{p}}_2^{(\infty,0)})$ .

Analogously we get that  $0 \in \partial_{V_2} F(\tilde{\mathbf{p}}_1^{(\infty,0)} + \cdot) (\mathbf{p}_2^{(\infty,1)})$ , i.e.,  $\mathbf{p}_2^{(\infty,1)} \in \arg \min_{\mathbf{p}_2 \in V_2} F(\tilde{\mathbf{p}}_1^{(\infty,0)} + \mathbf{p}_2)$ .

Moreover, by the monotone decrease of the energy we have that

$$2F(\mathbf{p}^{(k)}) \geq F(\mathbf{p}_1^{(k+1)} + \tilde{\mathbf{p}}_2^{(k)}) + F(\tilde{\mathbf{p}}_1^{(k)} + \mathbf{p}_2^{(k+1)}) \geq 2F(\mathbf{p}^{(k+1)}) \quad \text{for all } k \in \mathbb{N}$$

and hence

$$2(F(\mathbf{p}^{(k)}) - F(\mathbf{p}^{(k+1)})) \geq F(\mathbf{p}_1^{(k+1)} + \tilde{\mathbf{p}}_2^{(k)}) + F(\tilde{\mathbf{p}}_1^{(k)} + \mathbf{p}_2^{(k+1)}) - 2F(\mathbf{p}^{(k+1)}) \geq 0.$$

The sequence  $(\tilde{\mathbf{p}}_1^{(k_l)})_l$  is bounded and hence there exists a convergent subsequence  $(\tilde{\mathbf{p}}_1^{(k_{l_jm})})_m$  with limit  $\tilde{\mathbf{p}}_1^{(\infty,0)}$ . Moreover,  $(\mathbf{p}_2^{(k_{l_jm})})_m$  is bounded and has a convergent subsequence  $(\mathbf{p}_2^{(k_{l_jm_n})})_n$  with limit  $\mathbf{p}_2^{(\infty,0)}$ . Thus we have

$$\lim_{n \rightarrow \infty} \left[ F(\mathbf{p}_1^{(k_{l_jm_n}+1)} + \tilde{\mathbf{p}}_2^{(k_{l_jm_n})}) + F(\tilde{\mathbf{p}}_1^{(k_{l_jm_n})} + \mathbf{p}_2^{(k_{l_jm_n}+1)}) - 2F(\mathbf{p}^{(k_{l_jm_n+1})}) \right] = 0.$$

Since  $F(\mathbf{p}_1 + \tilde{\mathbf{p}}_2^{(\infty,0)}) \geq F(\mathbf{p}_1^{(\infty,1)} + \tilde{\mathbf{p}}_2^{(\infty,0)})$  for all  $\mathbf{p}_1 \in V_1$  and  $F(\tilde{\mathbf{p}}_1^{(\infty,0)} + \mathbf{p}_2) \geq F(\tilde{\mathbf{p}}_1^{(\infty,0)} + \mathbf{p}_2^{(\infty,1)})$  for all  $\mathbf{p}_2 \in V_2$  we deduce that

$$F(\tilde{\mathbf{p}}_1^{(\infty,0)} + \tilde{\mathbf{p}}_2^{(\infty,0)}) = F(\mathbf{p}_1^{(\infty,1)} + \tilde{\mathbf{p}}_2^{(\infty,0)}) \text{ and } F(\tilde{\mathbf{p}}_1^{(\infty,0)} + \tilde{\mathbf{p}}_2^{(\infty,0)}) = F(\tilde{\mathbf{p}}_1^{(\infty,0)} + \mathbf{p}_2^{(\infty,1)})$$

and hence

$$\tilde{\mathbf{p}}_1^{(\infty,0)} \in \arg \min_{\mathbf{p}_1 \in V_1} F(\mathbf{p}_1 + \tilde{\mathbf{p}}_2^{(\infty,0)}) \text{ and } \tilde{\mathbf{p}}_2^{(\infty,0)} \in \arg \min_{\mathbf{p}_2 \in V_2} F(\tilde{\mathbf{p}}_1^{(\infty,0)} + \mathbf{p}_2),$$

which concludes the proof.  $\square$

#### 4. Domain Decomposition for Dual Total Variation

**Theorem 4.8.** *The sequence  $(\mathbf{p}^{(k)})_k$  generated by Algorithm 2 has an accumulation point  $\mathbf{p}^{(\infty)} \in V$  which is a solution of the problem in (4.4).*

*Proof.* The proof can be carried out analogously to the proof of Theorem 4.4. However, we present an alternative proof.

By similar arguments as in Proposition 4.7 we extract suitable subsequences and due to their boundedness (Proposition 4.6), the hierarchy of sequences  $(\tilde{\mathbf{p}}_2^{(k_l)})_l$ ,  $(\mathbf{p}_1^{(k_{l_j}+1)})_j$ ,  $(\tilde{\mathbf{p}}_1^{(k_{l_{j_m}})})_m$ ,  $(\mathbf{p}_2^{(k_{l_{j_m}}+1)})_n$  converges to limit points  $\mathbf{p}_2^{(\infty,0)}$ ,  $\mathbf{p}_1^{(\infty,1)}$ ,  $\tilde{\mathbf{p}}_1^{(\infty,0)}$ ,  $\mathbf{p}_2^{(\infty,1)}$ , respectively. Hence the system

$$\begin{aligned}\mathbf{p}_1^{(k_{l_{j_m}}+1)} &\in \arg \min_{\mathbf{p}_1 \in V_1} \left\{ \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1 + \tilde{\mathbf{p}}_2^{(k_{l_{j_m}})}) + f\|_{\mathcal{H}}^2 : -\alpha \mathbf{1} \leq \mathbf{p}_1(x) + \tilde{\mathbf{p}}_2^{(k_{l_{j_m}})}(x) \leq \alpha \mathbf{1} \right\}, \\ \mathbf{p}_2^{(k_{l_{j_m}}+1)} &\in \arg \min_{\mathbf{p}_2 \in V_2} \left\{ \frac{1}{2} \|\operatorname{div}(\tilde{\mathbf{p}}_1^{(k_{l_{j_m}})} + \mathbf{p}_2) + f\|_{\mathcal{H}}^2 : -\alpha \mathbf{1} \leq \mathbf{p}_2(x) + \tilde{\mathbf{p}}_1^{(k_{l_{j_m}})}(x) \leq \alpha \mathbf{1} \right\},\end{aligned}$$

converges by [36, Theorem 24.4, p. 233] to

$$\begin{aligned}\mathbf{p}_1^{(\infty,1)} &\in \arg \min_{\mathbf{p}_1 \in V_1} \left\{ \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1 + \tilde{\mathbf{p}}_2^{(\infty,0)}) + f\|_{\mathcal{H}}^2 : -\alpha \mathbf{1} \leq \mathbf{p}_1(x) + \tilde{\mathbf{p}}_2^{(\infty,0)}(x) \leq \alpha \mathbf{1} \right\}, \\ \mathbf{p}_2^{(\infty,1)} &\in \arg \min_{\mathbf{p}_2 \in V_2} \left\{ \frac{1}{2} \|\operatorname{div}(\tilde{\mathbf{p}}_1^{(\infty,0)} + \mathbf{p}_2) + f\|_{\mathcal{H}}^2 : -\alpha \mathbf{1} \leq \mathbf{p}_2(x) + \tilde{\mathbf{p}}_1^{(\infty,0)}(x) \leq \alpha \mathbf{1} \right\},\end{aligned}$$

for  $n \rightarrow \infty$ .

By Proposition 4.7 we have that  $\mathbf{p}_2^{(\infty,1)}$  and  $\tilde{\mathbf{p}}_2^{(\infty,0)}$  are minimizers of

$$\begin{aligned}\min \frac{1}{2} \|\operatorname{div}(\tilde{\mathbf{p}}_1^{(\infty,0)} + \mathbf{p}_2) + f\|_{\mathcal{H}}^2 \quad \text{over } \mathbf{p}_2 \in V_2 \\ \text{s.t. } -\alpha \mathbf{1} \leq \mathbf{p}_2(x) + \tilde{\mathbf{p}}_1^{(\infty,0)}(x) \leq \alpha \mathbf{1},\end{aligned}$$

and  $\mathbf{p}_1^{(\infty,1)}$  and  $\tilde{\mathbf{p}}_1^{(\infty,0)}$  are minimizers of

$$\begin{aligned}\min \frac{1}{2} \|\operatorname{div}(\mathbf{p}_1 + \tilde{\mathbf{p}}_2^{(\infty,0)}) + f\|_{\mathcal{H}}^2 \quad \text{over } \mathbf{p}_1 \in V_1 \\ \text{s.t. } -\alpha \mathbf{1} \leq \mathbf{p}_1(x) + \tilde{\mathbf{p}}_2^{(\infty,0)}(x) \leq \alpha \mathbf{1},\end{aligned}$$

Hence we get the following first order conditions for  $\tilde{\mathbf{p}}_1^{(\infty,0)} \in V_1$  and  $\tilde{\mathbf{p}}_2^{(\infty,0)} \in V_2$ :

$$\left\langle \operatorname{div}(\tilde{\mathbf{p}}_1^{(\infty,0)} + \tilde{\mathbf{p}}_2^{(\infty,0)}) + f, \operatorname{div} \mathbf{q}_1 - \operatorname{div} \tilde{\mathbf{p}}_1^{(\infty,0)} \right\rangle_{\mathcal{H}} \geq 0$$

for all  $\mathbf{q}_1 \in V_1$ ,  $-\alpha \mathbf{1} \leq \mathbf{q}_1(x) + \tilde{\mathbf{p}}_2^{(\infty,0)}(x) \leq \alpha \mathbf{1}$  for all  $x \in \Omega$ .

$$\left\langle \operatorname{div}(\tilde{\mathbf{p}}_1^{(\infty,0)} + \tilde{\mathbf{p}}_2^{(\infty,0)}) + f, \operatorname{div} \mathbf{q}_2 - \operatorname{div} \tilde{\mathbf{p}}_2^{(\infty,0)} \right\rangle_{\mathcal{H}} \geq 0$$

for all  $\mathbf{q}_2 \in V_2$ ,  $-\alpha \mathbf{1} \leq \mathbf{q}_2(x) + \tilde{\mathbf{p}}_1^{(\infty,0)}(x) \leq \alpha \mathbf{1}$  for all  $x \in \Omega$ .

Then by adding up the last two inequalities we obtain

$$\left\langle \operatorname{div}(\tilde{\mathbf{p}}_1^{(\infty,0)} + \tilde{\mathbf{p}}_2^{(\infty,0)}) + f, \operatorname{div}(\mathbf{q}_1 + \mathbf{q}_2) - \operatorname{div}(\tilde{\mathbf{p}}_1^{(\infty,0)} + \tilde{\mathbf{p}}_2^{(\infty,0)}) \right\rangle_{\mathcal{H}} \geq 0$$

## I. Subspace Correction Methods for Total Variation Minimization

for all  $\mathbf{q}_i \in V_i$  with  $-\alpha\mathbf{1} \leq \mathbf{q}_1(x) + \tilde{\mathbf{p}}_2^{(\infty,0)}(x) \leq \alpha\mathbf{1}$  for all  $x \in \Omega$  and  $-\alpha\mathbf{1} \leq \mathbf{q}_2(x) + \tilde{\mathbf{p}}_1^{(\infty,0)}(x) \leq \alpha\mathbf{1}$  for all  $x \in \Omega$  for  $i = 1, 2$ . Since  $\Omega_1$  and  $\Omega_2$  are disjoint and  $-\alpha\mathbf{1} \leq \mathbf{q}_1(x) + \tilde{\mathbf{p}}_2^{(\infty,0)}(x) \leq \alpha\mathbf{1}$  as well as  $-\alpha\mathbf{1} \leq \mathbf{q}_2(x) + \tilde{\mathbf{p}}_1^{(\infty,0)}(x) \leq \alpha\mathbf{1}$  it follows that  $-\alpha\mathbf{1} \leq \mathbf{q}_1(x) + \mathbf{q}_2(x) \leq \alpha\mathbf{1}$  and  $-\alpha\mathbf{1} \leq \tilde{\mathbf{p}}_1^{(\infty,0)}(x) + \tilde{\mathbf{p}}_2^{(\infty,0)}(x) \leq \alpha\mathbf{1}$  for all  $x \in \Omega$ . Hence  $\mathbf{p}^{(\infty)} = \tilde{\mathbf{p}}_1^{(\infty,0)} + \tilde{\mathbf{p}}_2^{(\infty,0)} \in V$  is a solution of the global problem in (4.4).  $\square$

### 4.4. Implementation

In order to split the subdomain problems in Algorithm 1 and Algorithm 2 to the respective subdomains plus an arbitrary small stripe around the interface, we will require a certain splitting property of the discrete divergence operator with respect to a disjoint decomposition of the domain  $\Omega$ , i.e.,

$$\sum_{x \in \Omega} \operatorname{div}_{\Omega}(\mathbf{p}_1 + \mathbf{p}_2)(x) = \left( \sum_{x \in \Omega_1 \cup \widehat{\Omega}_2} \operatorname{div}_{\Omega_1 \cup \widehat{\Omega}_2}(\mathbf{p}_1 + \mathbf{p}_2)(x) \right) + d_1(\mathbf{p}_2)$$

$$\sum_{x \in \Omega} \operatorname{div}_{\Omega}(\mathbf{p}_1 + \mathbf{p}_2)(x) = \left( \sum_{x \in \Omega_2 \cup \widehat{\Omega}_1} \operatorname{div}_{\Omega_2 \cup \widehat{\Omega}_1}(\mathbf{p}_1 + \mathbf{p}_2)(x) \right) + d_2(\mathbf{p}_1)$$

where  $d_i$  is a suitable function only depending on  $\mathbf{p}_{i^c}$ ,  $i^c \in \{1, 2\} \setminus \{i\}$ , and  $\widehat{\Omega}_i \subset \Omega_i$  is a small stripe around the interface between  $\Omega_1$  and  $\Omega_2$  for  $i = 1, 2$ . A typical choice for  $\widehat{\Omega}_i$ ,  $i = 1, 2$ , for which this splitting property holds, is shown in Figure 4.1.



**Figure 4.1:** Nonoverlapping domain decomposition of  $\Omega$  into  $\Omega_1$  and  $\Omega_2$  with the small stripes  $\widehat{\Omega}_i \subset \Omega_i$  for  $i = 1, 2$ .

Let us consider, for example, the subspace minimization in  $V_1$ , i.e.,

$$\min \frac{1}{2} \|\operatorname{div}_{\Omega}(\mathbf{p}_1 + \mathbf{p}_2^{(k)}) + f\|_{\mathcal{H}}^2 \quad \text{over } \mathbf{p}_1 \in V_1$$

$$\text{s.t. } -\alpha\mathbf{1} \leq \mathbf{p}_1(x) \leq \alpha\mathbf{1} \text{ for all } x \in \Omega_1,$$

#### 4. Domain Decomposition for Dual Total Variation

which can be written as

$$\begin{aligned} & \min \frac{1}{2} \|\operatorname{div}_{\Omega_1 \cup \widehat{\Omega}_2}(\mathbf{p}_1 + \mathbf{p}_2^{(k)}) + f + d_1(\mathbf{p}_2^{(k)})\|_{\ell^2(\Omega_1 \cup \widehat{\Omega}_2)}^2 \quad \text{over } \mathbf{p}_1 \in V_1 \\ & \text{s.t. } -\alpha \mathbf{1} \leq \mathbf{p}_1(x) \leq \alpha \mathbf{1} \text{ for all } x \in \Omega_1, \end{aligned} \quad (4.9)$$

where we used the above splitting property of the divergence operator. Note that the inequality constraint is considered in  $\Omega_1$  only, which is due to the fact that  $\mathbf{p}_1 \in V_1$  and hence  $0 = \mathbf{p}_1(x) \in [-\alpha, \alpha]$  for all  $x \in \Omega \setminus \Omega_1$  automatically; compare with the definition of  $V_1$ .

We define  $\widehat{V}_2 := \{\mathbf{v} \in V : \mathbf{v}(x) = \mathbf{0} \text{ for } x \in \Omega \setminus \widehat{\Omega}_2\}$ ,  $\|\mathbf{p}\|_{V_1 \oplus \widehat{V}_2}^2 := h^2 \sum_{x \in \Omega_1 \cup \widehat{\Omega}_2} |p^1(x)|^2 + |p^2(x)|^2$  for all  $\mathbf{p} = (p^1, p^2) \in V_1 \oplus \widehat{V}_2$ ,  $\langle \mathbf{p}, \mathbf{q} \rangle_{V_1 \oplus \widehat{V}_2} := h^2 \sum_{x \in \Omega_1 \cup \widehat{\Omega}_2} \langle \mathbf{p}(x), \mathbf{q}(x) \rangle_{\mathbb{R}^2}$  for all  $\mathbf{p}, \mathbf{q} \in V_1 \oplus \widehat{V}_2$  and rewrite the constrained minimization problem (4.9) in the following from

$$\begin{aligned} & \min \frac{1}{2} \|\operatorname{div}_{\Omega_1 \cup \widehat{\Omega}_2}(\xi_1) + f + d_1(\mathbf{p}_2^{(k)})\|_{\ell^2(\Omega_1 \cup \widehat{\Omega}_2)}^2 \quad \text{over } \xi_1 \in V_1 \oplus \widehat{V}_2 \\ & \text{s.t. } \pi_{\widehat{V}_2} \xi_1 = \pi_{\widehat{V}_2} \mathbf{p}_2^{(k)} \text{ and } -\alpha \mathbf{1} \leq \xi_1(x) \leq \alpha \mathbf{1} \text{ for all } x \in \Omega_1 \cup \widehat{\Omega}_2. \end{aligned}$$

Note that for  $i = 1, 2$  and  $i^c = \{1, 2\} \setminus \{i\}$  indeed  $\xi_i$  is optimal if and only if  $\mathbf{p}_i = \xi_i - \mathbf{p}_{i^c}$  is optimal. In order to solve this constrained minimization problem we use the *augmented Lagrangian method* [3, 30]:

**Algorithm 3.**

- Choose  $\xi_1^{(0)} (= \pi_{V_1 \oplus \widehat{V}_2} \mathbf{p}^{(k)}) \in V_1$ ,  $\mathbf{q}^{(0)} (= \mathbf{0}) \in V_1$ , and  $\mu > 0$
- Iterate: For  $n = 0, 1, \dots$

$$\begin{aligned} \xi_1^{(n+1)} \in \arg \min_{\xi_1 \in V_1 \oplus \widehat{V}_2} & \left\{ \frac{1}{2} \|\operatorname{div}_{\Omega_1 \cup \widehat{\Omega}_2}(\xi_1) + f + d_1(\mathbf{p}_2^{(k)})\|_{\ell^2(\Omega_1 \cup \widehat{\Omega}_2)}^2 - \langle \mathbf{q}^{(n)}, \xi_1 - \xi_1^{(n)} \rangle_{V_1 \oplus \widehat{V}_2} \right. \\ & \left. + \frac{\mu}{2} \|\pi_{\widehat{V}_2}(\xi_1 - \mathbf{p}_2^{(k)})\|_{V_1 \oplus \widehat{V}_2}^2 : -\alpha \mathbf{1} \leq \xi_1(x) \leq \alpha \mathbf{1} \right\} \end{aligned}$$

$$\mathbf{q}^{(n+1)} := \mathbf{q}^{(n)} - \mu \pi_{\widehat{V}_2}^* \pi_{\widehat{V}_2} (\xi_1^{(n+1)} - \mathbf{p}_2^{(k)})$$

Note that Algorithm 3 penalizes the constraint  $\pi_{\widehat{V}_2} \xi_1 = \pi_{\widehat{V}_2} \mathbf{p}_2^{(k)}$  by adding a corresponding term to the objective functional. The associated penalty parameter is  $\mu > 0$ . This results in an optimization problem with bilateral constraints in the second step of the algorithm. This minimization problem can be treated as proposed in [25, Section 3], where a family of approximating problems is considered. This method has the advantage of an image resolution independent convergence for fixed  $\mu$  and it converges locally at a superlinear rate. More precisely, the minimization problem is replaced by a regularized problem, i.e., we consider

$$\begin{aligned} & \min \frac{1}{2\beta} |\nabla \xi_1|^2 + \frac{1}{2} \|\operatorname{div}_{\Omega_1 \cup \widehat{\Omega}_2}(\xi_1) + f + d_1(\mathbf{p}_2^{(k)})\|_{\ell^2(\Omega_1 \cup \widehat{\Omega}_2)}^2 + \frac{\gamma}{2} \|P_{div} \xi_1\|_{V_1 \oplus \widehat{V}_2}^2 \\ & + \frac{1}{2\beta} \|\max(0, \beta(\xi_1 - \alpha \mathbf{1}))\|_{V_1 \oplus \widehat{V}_2}^2 + \frac{1}{2\beta} \|\min(0, \beta(\xi_1 + \alpha \mathbf{1}))\|_{V_1 \oplus \widehat{V}_2}^2 \\ & - \langle \mathbf{q}^{(n)}, \xi_1 - \xi_1^{(n)} \rangle_{V_1 \oplus \widehat{V}_2} + \frac{\mu}{2} \|\pi_{\widehat{V}_2}(\xi_1 - \mathbf{p}_2^{(k)})\|_{V_1 \oplus \widehat{V}_2}^2 \quad \text{over } \xi_1 \in V_1 \oplus \widehat{V}_2, \end{aligned} \quad (4.10)$$

## I. Subspace Correction Methods for Total Variation Minimization

where  $\beta > 0$ ,  $\gamma \geq 0$ ,  $\nabla$  denotes the discrete vector gradient in  $\Omega_1 \cup \widehat{\Omega}_2$ ,  $|\cdot|$  is an arbitrary norm in  $(V_1 \oplus \widehat{V}_2)^2$ , and  $P_{div}$  is the orthogonal projection in  $V_1 \oplus \widehat{V}_2$  onto  $\{\mathbf{v} \in V_1 \oplus \widehat{V}_2 : \operatorname{div}_{\Omega_1 \cup \widehat{\Omega}_2} \mathbf{v}(x) = 0 \text{ for all } x \in \Omega_1 \cup \widehat{\Omega}_2\}$ . The term  $\frac{1}{2\beta} |\nabla \xi_1|^2$  is added to account for function space regularity, which is needed for guaranteeing that the max- and min-operations are semismooth (in function space) in the first-order system associated with (4.10). As a consequence, for  $\beta > 0$  the semismooth Newton iteration exhibits a (image) resolution independent convergence. As argued in [25], the parameter  $\gamma$  has no noticeable effect on the result and thus can be set to 0, as it is done in our numerical experiments below. Let  $\xi_1^*$  denote the unique solution to (4.10). Then it satisfies the following optimality conditions

$$\begin{aligned} -\frac{1}{\beta} \Delta \xi_1^* - \nabla_{\Omega_1 \cup \widehat{\Omega}_2} (\operatorname{div}_{\Omega_1 \cup \widehat{\Omega}_2} \xi_1^* + f + d_1(\mathbf{p}_2^{(k)})) + \gamma P_{div} \xi_1^* - \mathbf{q}^{(n)} + \mu \pi_{\widehat{V}_2}^* \pi_{\widehat{V}_2} (\xi_1^* - \mathbf{p}_2^{(k)}) + \lambda &= 0 \\ \lambda &= \max(0, \beta(\xi_1^* - \alpha \mathbf{1})) + \min(0, \beta(\xi_1^* + \alpha \mathbf{1})), \end{aligned} \quad (4.11)$$

where  $\Delta$  is the discrete vector Laplacian in  $\Omega_1 \cup \widehat{\Omega}_2$ . This leads to the following active-set algorithm for computing  $\xi_1^{(n+1)}$ , cf. [25, Algorithm B],

**Algorithm 4** (active-set algorithm).

- *Initialize:*  $\zeta_{(0)} = \xi_1^{(n)}$
- *Iterate:* Set  $\ell = 0$ . While stopping rule not satisfied do:
  - Set, for  $j = 1, 2$ ,

$$\begin{aligned} \mathcal{A}_{\ell+1}^{+,j} &= \{x : \zeta_{(\ell)}^j(x) - \alpha > 0\}, \\ \mathcal{A}_{\ell+1}^{-,j} &= \{x : \zeta_{(\ell)}^j(x) + \alpha < 0\}, \\ \mathcal{I}_{\ell+1}^j &= (\Omega_1 \cup \widehat{\Omega}_2) \setminus (\mathcal{A}_{\ell+1}^{+,j} \cup \mathcal{A}_{\ell+1}^{-,j}) \end{aligned}$$

– Solve for  $\zeta_{(\ell+1)}$

$$\begin{aligned} \left( -\frac{1}{\beta} \Delta - \nabla_{\Omega_1 \cup \widehat{\Omega}_2} \operatorname{div}_{\Omega_1 \cup \widehat{\Omega}_2} + \gamma P_{div} \right) \zeta_{(\ell+1)} + \mu \pi_{\widehat{V}_2}^* \pi_{\widehat{V}_2} (\zeta_{(\ell+1)} - \mathbf{p}_2^{(k)}) + \\ \beta(\zeta_{(\ell+1)} - \alpha \mathbf{1}) \chi_{\mathcal{A}_{\ell+1}^{+,j}} + \beta(\zeta_{(\ell+1)} + \alpha \mathbf{1}) \chi_{\mathcal{A}_{\ell+1}^{-,j}} - \\ \nabla_{\Omega_1 \cup \widehat{\Omega}_2} \operatorname{div}_{\Omega_1 \cup \widehat{\Omega}_2} (f + d_1(\mathbf{p}_2^{(k)})) - \mathbf{q}^{(n)} &= 0. \end{aligned} \quad (4.12)$$

–  $\ell := \ell + 1$ .

- Set  $\xi_1^{(n+1)} = \zeta_{(\ell)}$ .

In the above algorithm we used  $\chi_{\mathcal{A}_{\ell+1}^{\pm}} = (\chi_{\mathcal{A}_{\ell+1}^{\pm}}^1, \chi_{\mathcal{A}_{\ell+1}^{\pm}}^2)$  with

$$\chi_{\mathcal{A}_{\ell+1}^{\pm}}^j = \begin{cases} 1 & \text{if } x \in \mathcal{A}_{\ell+1}^{\pm,j}, \\ 0 & \text{if } x \notin \mathcal{A}_{\ell+1}^{\pm,j}, \end{cases}$$

for  $j = 1, 2$ . A few words on Algorithm 4 are in order: The active and inactive set determination in the second step results from a generalized differentiation of the system (4.11) at

#### 4. Domain Decomposition for Dual Total Variation

$\zeta_{(\ell)}^j$ ,  $j = 1, 2$ , utilizing a specific element of the set-valued generalized derivative. For more details on this step we refer to [25, Algorithm B]. The third step corresponds to the linear system, which needs to be solved at every iteration of the generalized Newton scheme. The while-loop terminates when the residual of the respective linear system (4.12) drops below a user-specified tolerance for the first time. This tolerance is given in iteration  $\ell + 1$  by  $\text{tol}_{\ell+1} = 0.1 \min(r_\ell^{1.25}, r_\ell)$ , where  $r_\ell$  denotes the residual in the  $\ell$ -th iteration, as suggested in [25]. The linear system in Algorithm 4 is resolved by the conjugate gradient method (CG-method). The stopping tolerance for the CG-method in iteration  $\ell$  is again the aforementioned tolerance  $\text{tol}_\ell$ .

We conclude this section by mentioning that the implementation for the minimization in  $V_2$  works analogously by simply adjusting the notations accordingly.

### 4.5. Numerical Examples

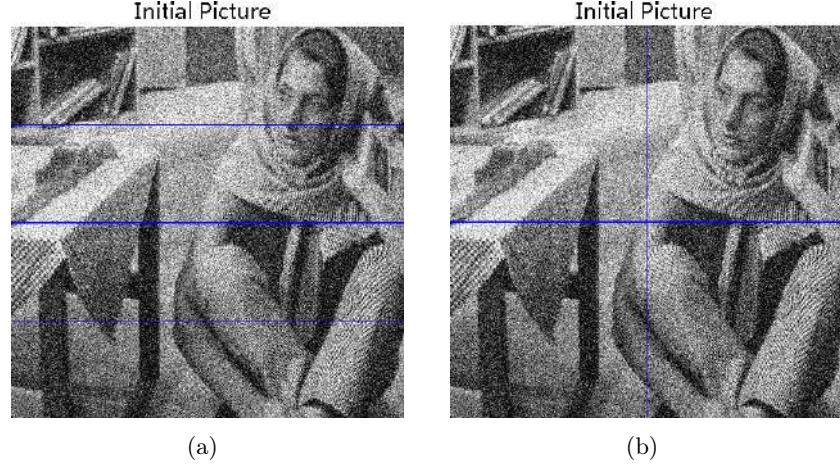
In this section we present numerical experiments for the proposed sequential and parallel algorithms for image denoising. The values of the parameters are chosen experimentally, i.e., we choose the values which give a good compromise between visual quality and computational time of the algorithm. Note that the dynamic range of all image data considered in this paper is  $[0, 1]$ . All the computations presented are done in Matlab on a Linux cluster with 32 kernels, where each kernel has 2 processors and each processor 4 cores, i.e., on a computer with 256 cores.

#### 4.5.1. Denoising of 2D Image Data

In our numerical experiments for 2D image denoising we test Algorithm 1 and Algorithm 2 for different images and numbers of subdomains ( $D = 2, 4, 8, 16$ ) and compare their performance with each other and with the algorithm in [25, Algorithm B] without decomposing the domain ( $D = 1$ ). We terminate the algorithms as soon as the residual of the global regularized problem, cf. (4.10), drops below a certain threshold  $R^* > 0$ .

We consider the “barbara” image of size  $512 \times 512$  pixels, depicted in Figure 4.2, which is corrupted by Gaussian white noise with variance 0.03. The sequential and parallel domain decomposition algorithms are applied to this image by decomposing the image domain into  $D = 2, 4, 8, 16$  subdomains. In particular, we either split the physical domain into stripes or windows as shown in Figure 4.2. In the algorithms we choose  $\alpha = 0.13$ ,  $\beta = 7$ ,  $\gamma = 0$ , and  $\mu = 500$ . The iterations are terminated as soon as the global residual reaches the significance value of  $R^* = 2.1 \times 10^{-4}$ . We note that  $R^*$  was chosen empirically as the quality of the reconstruction was satisfactory and no noticeable change occurred for smaller values of  $R^*$ . For different numbers of splittings we show in Table 4.1 the required computational time and the number of required iterations. Moreover, in Table 4.1 we also display the approximate average time for one (outer) iteration of Algorithm 1, Algorithm 2 or [25, Algorithm B], respectively. Since the image size is very small the algorithm without decomposition converges fastest. However, when the image size increases a decomposition becomes essential, as we see in Section 4.5.3 for the example of 3D medical imaging, where the domain decomposition algorithms outperform the global method. Moreover, due to limitation of hardware, a decomposition strategy is anyway necessary when the entire problem is so huge that it cannot be solved without splitting into smaller problems. In the rest of this section we concentrate on a performance-comparison

## I. Subspace Correction Methods for Total Variation Minimization



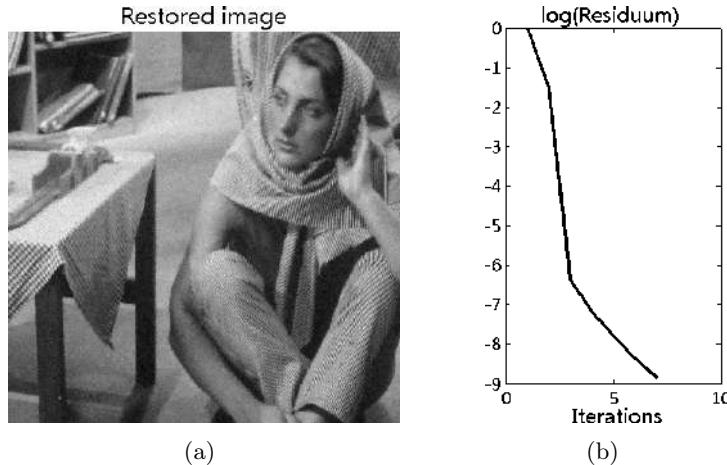
**Figure 4.2:** Domain decomposition in 4 stripes, see (a), and 4 windows, see (b), of the “barbara” image corrupted by Gaussian noise with standard deviation 0.03.

# domains	sequential	parallel
$D = 1$ :	10 s / 10 it (1 s/it)	
$D = 2$ (stripe):	124 s / 6 it (20 s/it)	262 s / 16 it (16 s/it)
$D = 4$ (stripe):	145 s / 7 it (20 s/it)	231 s / 19 it (12 s/it)
$D = 4$ (window):	150 s / 7 it (21 s/it)	236 s / 22 it (10 s/it)
$D = 8$ (stripe):	173 s / 8 it (21 s/it)	253 s / 22 it (11 s/it)
$D = 16$ (stripe):	235 s / 10 it (23 s/it)	353 s / 26 it (13 s/it)
$D = 16$ (window):	180 s / 8 it (22 s/it)	302 s / 26 it (11 s/it)

**Table 4.1:** Restoration of the “barbara” image in Figure 4.2 of size  $512 \times 512$  pixels: Computational performance (CPU time in seconds / the number of iterations and the average CPU time for one iteration in Algorithm 1, Algorithm 2 and [25, Algorithm B], respectively) for the sequential and parallel domain decomposition algorithms and [25, Algorithm B] with  $\alpha = 0.13$ ,  $\beta = 7$ ,  $\gamma = 0$ ,  $\mu = 500$  for different numbers of subdomains ( $D = 1, 2, 4, 8, 16$ ). The algorithms are stopped as soon as the global residual reaches the significance level  $2.1 \times 10^{-4}$ .

#### 4. Domain Decomposition for Dual Total Variation

between the proposed sequential and parallel algorithms. From Table 4.1 we observe that the number of iterations is increasing with the number of subdomains. This phenomenon may be attributed to the increasing number of interfaces, where the correct solution can only be obtained during the iterations, while inside the domains the solution seems to be immediately achieved. Due to the fact that the image size is rather small, parallelization does not pay off, since the communication time between the processors adds to the CPU-time and more iterations are needed until convergence is reached. The latter fact comes from the averaging of the current iterate with the previous one, cf. Algorithm 2. Nevertheless we emphasize that for each single iteration significantly less computational time is needed with the parallel version than with the sequential one. We also note, that we ran experiments in which we modified the parallel algorithm in a way that the update is done without any averaging, i.e.,  $\mathbf{p}^{(k+1)} = \sum_{i=1}^D \mathbf{p}_i^{(k+1)}$ . Additionally to the fact that analytically we cannot proof convergence with this modified update, we observed in our numerical experiments that convergence to a global minimizer cannot be guaranteed in general. Thus the averaging process in the update is not only theoretically but also practically necessary.



**Figure 4.3:** In (a) we depict the restoration of the “barbara” image in Figure 4.2. For the “barbara” image of size  $2024 \times 2024$  pixels the decrease of the logarithm of the global residual for the sequential domain decomposition with 4 subdomains (stripes) is plotted in (b).

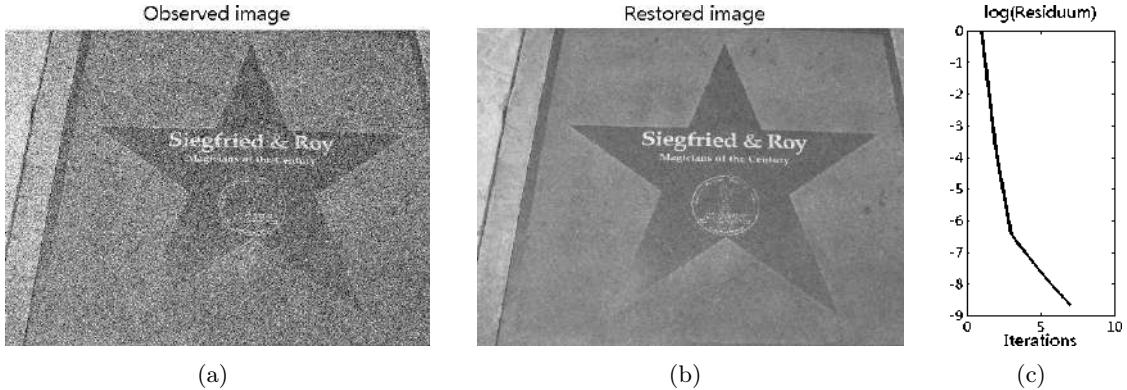
Since an image of size  $512 \times 512$  pixels is rather small, for the sake of comparison we rescaled the “barbara” image to a size of  $2048 \times 2048$  pixels and denoised it by the sequential and parallel algorithm with the same parameters and stopping criterion as used above. The results of these experiments are documented in Table 4.2. For a decomposition into 4 stripes the restored image is plotted in Figure 4.3 (a), while in Figure 4.3 (b) the progress of the global residual over the iterations is depicted. Still, the sequential algorithm is faster than its parallel version, which can be explained as before. However, for a decomposition into 8 stripes the parallel domain decomposition algorithm outperforms the sequential one, although more than twice as many iterations are needed. Additionally we observe that the number of domains for which the algorithm gives the best performance clearly depends on the size of the considered image. While for the small size “barbara” image the best performance is achieved for a splitting into two domains for the sequential algorithm and into four domains for the

## I. Subspace Correction Methods for Total Variation Minimization

# domains	sequential	parallel
$D = 2$ (stripe):	2736 s / 5 it (547 s/it)	4580 s / 13 it (352 s/it)
$D = 4$ (stripe):	2475 s / 6 it (412 s/it)	2898 s / 16 it (181 s/it)
$D = 4$ (window):	2209 s / 5 it (441 s/it)	3407 s / 17 it (200 s/it)
$D = 8$ (stripe):	2980 s / 7 it (425 s/it)	2292 s / 19 it (120 s/it)
$D = 16$ (stripe):	2852 s / 7 it (407 s/it)	2825 s / 22 it (128 s/it)
$D = 16$ (window):	2435 s / 6 it (405 s/it)	2902 s / 21 it (138 s/it)

**Table 4.2:** Restoration of the “barbara” image in Figure 4.2 rescaled to size  $2048 \times 2048$  pixels: Computational performance (CPU time in seconds / the number of iterations and the average CPU time for one iteration in Algorithm 1 and Algorithm 2 respectively) for the sequential and parallel domain decomposition algorithms with  $\alpha = 0.13$ ,  $\beta = 7$ ,  $\gamma = 0$ ,  $\mu = 500$  for different numbers of subdomains ( $D = 2, 4, 8, 16$ ). The algorithms are stopped as soon as the global residual reaches the significance level  $2.1 * 10^{-4}$ .

parallel version, for the rescaled image four domains for the sequential and eight domains for the parallel algorithm converge fastest. We further note that the CPU-cost per iteration exhibits a stronger increase for the sequential algorithm when compared with the parallel one. This indicates that for yet larger image sizes a trade-off between the sequential and the parallel method will be reached rendering the parallel version more efficient for growing image data sizes. This claim is substantiated in Section 4.5.3 for the example of 3D medical imaging where the parallel method is indeed more efficient than the sequential method.



**Figure 4.4:** Domain decomposition of an image (size  $1920 \times 2576$  pixels) corrupted by Gaussian noise with variance 0.03, see (a). In (b) we show the restored image, whereby we used the sequential domain decomposition algorithm for 4 domains with the following parameters:  $\alpha = 0.13$ ,  $\beta = 7$ ,  $\gamma = 0$ ,  $\mu = 500$ . In (c) we depict the progress of the logarithm of the global residual.

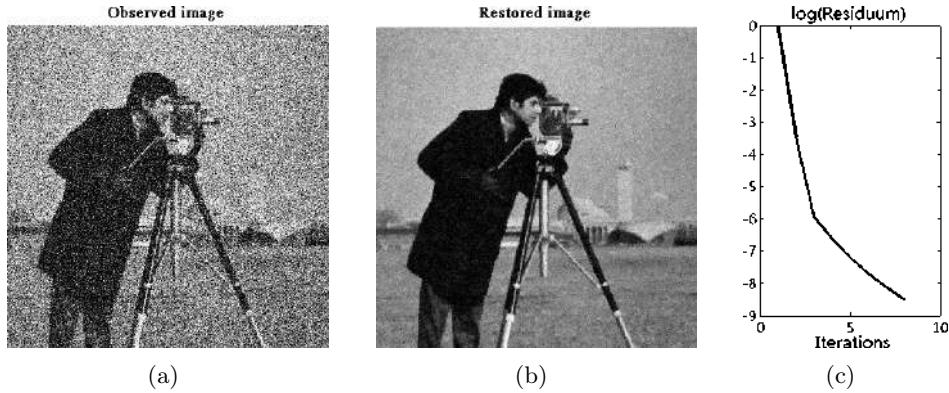
We also tested the sequential and parallel domain decomposition algorithm for the image in Figure 4.4, which is of size  $1920 \times 2576$  pixels corrupted by Gaussian white noise with variance 0.03, and for the image in Figure 4.5 of size  $1280 \times 1280$  pixels corrupted by Gaussian noise

#### 4. Domain Decomposition for Dual Total Variation

# domains	sequential	parallel
$D = 2$ (stripe):	2896 s / 5 it (579 s/it)	5374 s / 14 it (383 s/it)
$D = 4$ (stripe):	2675 s / 6 it (445 s/it)	3262 s / 18 it (181 s/it)
$D = 4$ (window):	2848 s / 6 it (474 s/it)	3535 s / 17 it (207 s/it)
$D = 8$ (stripe):	3114 s / 7 it (444 s/it)	3191 s / 21 it (151 s/it)
$D = 16$ (stripe):	3673 s / 8 it (459 s/it)	3786 s / 23 it (164 s/it)
$D = 16$ (window):	3267 s / 7 it (466 s/it)	3385 s / 22 it (153 s/it)

**Table 4.3:** Restoration of the image in Figure 4.4 of size  $1920 \times 2576$  pixels: Computational performance (CPU time in seconds / the number of iterations and the average CPU time for one iteration in Algorithm 1 and Algorithm 2 respectively) for the sequential and parallel domain decomposition algorithms with  $\alpha = 0.13$ ,  $\beta = 7$ ,  $\gamma = 0$ ,  $\mu = 500$  for different numbers of subdomains ( $D = 2, 4, 8, 16$ ). The algorithms are stopped as soon as the global residual reaches the significance level  $2.1 * 10^{-4}$ .

with variance 0.1. The results are reported in Table 4.3 and Table 4.4. For the restoration of the image in Figure 4.4 we set the parameters in the algorithms to  $\alpha = 0.13$ ,  $\beta = 7$ ,  $\gamma = 0$ ,  $\mu = 500$ , while for the one in Figure 4.5 we choose  $\alpha = 0.25$ ,  $\beta = 5$ ,  $\gamma = 0$ ,  $\mu = 500$ . Upon inspection of the figures in the tables, similar conclusions as for the previous test examples can be drawn.



**Figure 4.5:** Domain decomposition of an image (size  $1280 \times 1280$  pixels) corrupted by Gaussian noise with variance 0.1, see (a). In (b) we show the restored image, whereby we used the sequential domain decomposition algorithm for 4 domains with the following parameters:  $\alpha = 0.25$ ,  $\beta = 5$ ,  $\gamma = 0$ ,  $\mu = 500$ . In (c) we depict the progress of the logarithm of the global residual.

Since for image denoising the regularization parameter  $\alpha$  is usually small, we test our sequential domain decomposition algorithm also for relatively large  $\alpha$ . Such a choice of  $\alpha$  is relevant in scale separation in images. The associated imaging task is to identify features at different scales. For instance, a large  $\alpha$  produces a cartoon-like reconstruction, whereas small values of  $\alpha$  yield small scale features. In our test we choose  $\alpha = 1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}$  for separating

## I. Subspace Correction Methods for Total Variation Minimization

# domains	sequential	parallel
$D = 2$ (stripe):	1207 s / 6 it (201 s/it)	2705 s / 20 it (135 s/it)
$D = 4$ (stripe):	1108 s / 7 it (158 s/it)	1640 s / 24 it (68 s/it)
$D = 4$ (window):	1210 s / 7 it (172 s/it)	1801 s / 22 it (81 s/it)
$D = 8$ (stripe):	1326 s / 9 it (147 s/it)	1309 s / 27 it (48 s/it)
$D = 16$ (stripe):	1506 s / 10 it (150 s/it)	1682 s / 31 it (54 s/it)
$D = 16$ (window):	1199 s / 8 it (149 s/it)	1590 s / 28 it (56 s/it)

**Table 4.4:** Restoration of the image in Figure 4.5 of size  $1280 \times 1280$  pixels: Computational performance (CPU time in seconds / the number of iterations and the average CPU time for one iteration in Algorithm 1 and Algorithm 2 respectively) for the sequential and parallel domain decomposition algorithms with  $\alpha = 0.25$ ,  $\beta = 5$ ,  $\gamma = 0$ ,  $\mu = 500$  for different numbers of subdomains ( $D = 2, 4, 8, 16$ ). The algorithms are stopped as soon as the global residual reaches the significance level  $2.1 * 10^{-4}$ .

# domains \ \alpha	$\alpha = 1$	$\alpha = \frac{1}{2}$	$\alpha = \frac{1}{4}$	$\alpha = \frac{1}{8}$	$\alpha = \frac{1}{16}$
$D = 2$ (stripe):	1626 s / 17 it	897 s / 12 it	378 s / 9 it	126 s / 6 it	56 s / 4 it
$D = 4$ (stripe):	2087 s / 24 it	1105 s / 16 it	450 s / 10 it	148 s / 7 it	66 s / 5 it
$D = 8$ (stripe):	2589 s / 32 it	1282 s / 20 it	593 s / 13 it	173 s / 8 it	69 s / 5 it

**Table 4.5:** Scale separation of the “barbara” image in Figure 4.2 of size  $512 \times 512$  pixels: Computational performance for different regularization parameters  $\alpha$  (CPU time in seconds / the number of iterations) for the sequential domain decomposition algorithm with  $\beta = 7$ ,  $\gamma = 0$ ,  $\mu = 500$  for different numbers of subdomains ( $D = 2, 4, 8$ ). The algorithms are stopped as soon as the global residual reaches the significance level  $2.1 * 10^{-4}$ .

the scales of the image in Figure 4.2. In Table 4.5 we report on the behavior of the algorithm and observe that for large values of  $\alpha$  the domain decomposition algorithm slows down dramatically. This behavior can be attributed to the fact that large  $\alpha$  admits large variations of the solution  $\mathbf{p}$  of the dual problem (4.3). In particular, the size of the active-set (in  $\Omega$ ), i.e. the set where components of  $|\mathbf{p}(x)|$ ,  $x \in \Omega$ , coincide with  $\alpha$ , decreases as  $\alpha$  increases. In terms of the primal problem (4.2) this means that the size of the subset of  $\Omega$ , where the nondifferentiability of the regularization term becomes relevant, increases with  $\alpha$ .

### 4.5.2. Comparison of the Dual and Primal Domain Decomposition Approach

In [26] domain decomposition methods directly applied to the minimization of the primal problem (4.2) are introduced, which we refer to as the primal domain decomposition approach. In fact, we only compare the respective sequential algorithms, i.e., Algorithm 1 with the algorithm in [26, (3.1)], since a similar behaviour can be observed from the parallel versions. The comparison is done with respect to CPU-time and iterations for the restoration of the “barbara” image of size  $512 \times 512$  pixels corrupted by Gaussian white noise with variance

#### 4. Domain Decomposition for Dual Total Variation

# domains	dual	primal
$D = 2 :$	233 s / 29 it (8 s/it)	344 s / 136 it (2.5 s/it)
$D = 3 :$	168 s / 30 it (5.6 s/it)	3957 s / 500 it (7.9 s/it)
$D = 4 :$	113 s / 29 it (3.8 s/it)	5199 s / 500 it (10.3 s/it)
$D = 5 :$	97 s / 29 it (3.3 s/it)	6032 s / 500 it (12 s/it)
$D = 6 :$	92 s / 29 it (3.1 s/it)	6725 s / 500 it (13.4 s/it)
$D = 7 :$	92 s / 28 it (3.2 s/it)	6733 s / 500 it (13.4 s/it)
$D = 8 :$	89 s / 28 it (3.1 s/it)	7265 s / 500 it (14.5 s/it)

**Table 4.6:** Comparison of the proposed dual domain decomposition Algorithm 1 and the primal domain decomposition algorithms introduced in [26] for the restoration of the image in Figure 4.2 of size  $512 \times 512$  pixels: Computational performance (CPU time in seconds / the number of iterations and the average CPU time for one iteration) for different numbers of subdomains ( $D = 2, 3, 4, 5, 6, 7, 8$ ). The algorithms are stopped as soon as the global primal energy in (4.2) drops below 0.0645388 or the number of iterations reaches 500 first.

0.03. The physical domain is split into different numbers of stripes ( $D = 2, 3, 4, 5, 6, 7, 8$ ) as depicted in Figure 4.2(a). For the restoration we choose  $\alpha = 1/3$  in (4.2) and (4.3). Further, in Algorithm 1 we set  $\beta = 70$ ,  $\gamma = 0$ ,  $\mu = 500$ , while in the primal domain decomposition algorithm we set  $\gamma = 0.01$ ,  $\mu = 1$ , and  $\delta = 0.99$  as proposed in [26]. As a stopping criterion we use the energy in (4.2) which is at hand for both approaches, since for a solution  $\mathbf{p}^*$  of (4.3) and a minimizer  $u^*$  of (4.2) we have  $u^* = \operatorname{div} \mathbf{p}^* + f$ ; see [25]. For obtaining an appropriate stopping value for our tests we first solve the dual domain decomposition algorithm with a certain accuracy, i.e., we terminate the iterations as soon as the global residual reaches the significance value  $R^* = 10^{-2}$ . Then the observed primal energy is taken to be the stopping value, i.e., we stop the algorithms as soon as the primal energy drops below 0.0645388 for the first time in our numerical experiments. While the dual domain decomposition algorithm reaches the prescribed energy after a few iterations, the computation by the primal domain decomposition method takes many more iterations or does not even reach the desired value within 500 iterations; see Table 4.6. Since only a few iterations are needed for the dual algorithm, it converges much faster in CPU-time than the primal domain decomposition approach. This behaviour makes the dual splitting strategy superior for large-scale problems as they may occur in 3D imaging.

##### 4.5.3. Denoising 3D Computerized Tomography (CT) Images

We test Algorithm 1 and Algorithm 2 for denoising 3D medical data of  $512 \times 512 \times 48$  voxels, i.e., the data consists of 48 slices with each slice containing  $512 \times 512$  pixels; see Figure 4.6 for a vertical cut through the restoration. Solving this restoration task by minimizing (4.3) involves a huge computational effort. Therefore it is sensible to split the computation load into smaller pieces by using a domain decomposition strategy. A perhaps too simple splitting approach would be, minimizing (4.3) on each slice separately, i.e., instead of solving the whole 3D problem we solve 48 2D problems. This strategy only finds minimizers of each slice but does not find the minimizer of the associated 3D problem. Moreover, it is clear

## I. Subspace Correction Methods for Total Variation Minimization

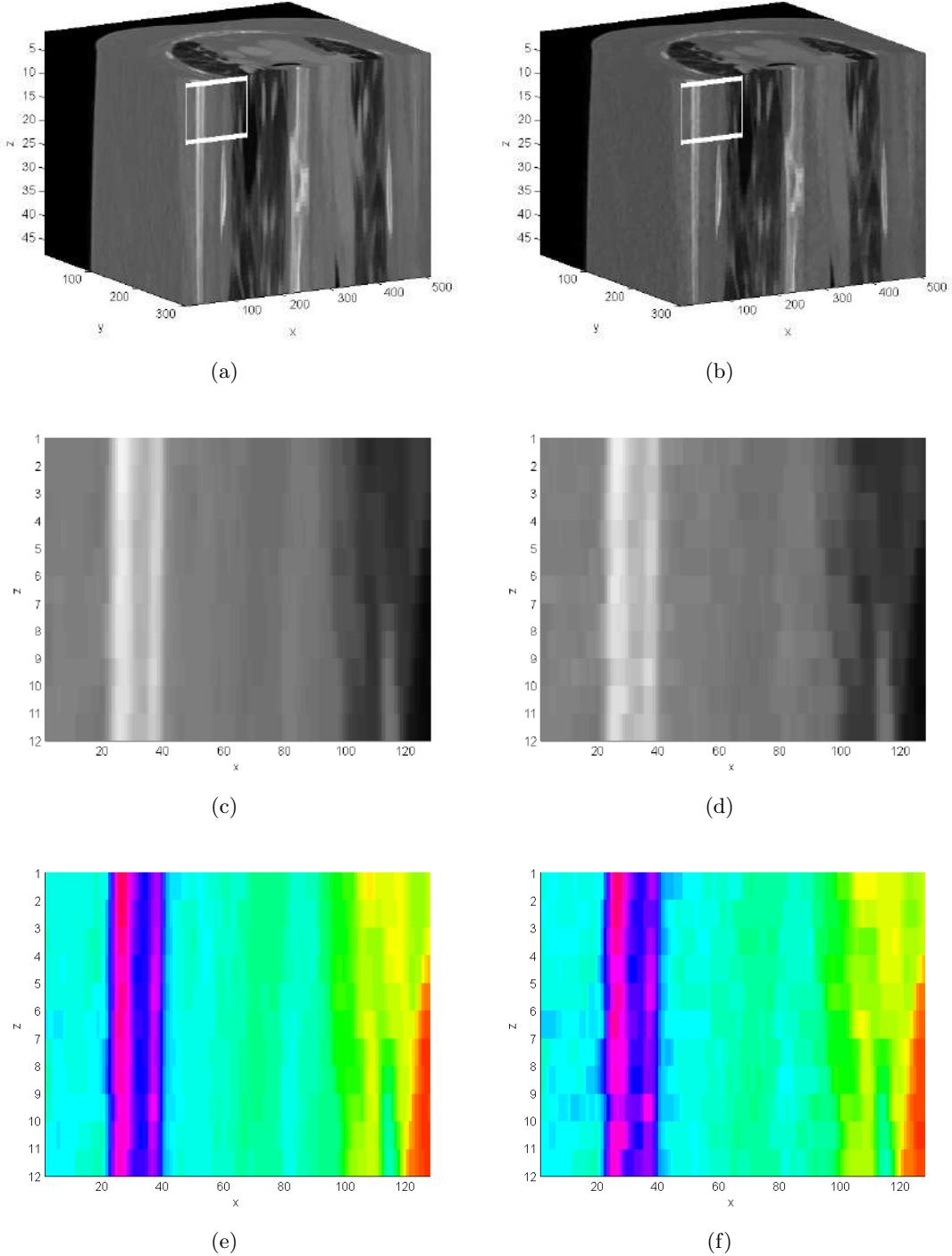
# domains	parallel	sequential
$D = 1 :$	10505 s / 7 it	
$D = 2 :$	9791 s / 6 it	9758 s / 4 it
$D = 4 :$	7123 s / 7 it	8238 s / 4 it
$D = 8 :$	5903 s / 7 it	7569 s / 4 it

**Table 4.7:** Comparison of the proposed sequential and parallel domain decomposition algorithms and [25, Algorithm B] with  $\alpha = 0.013$ ,  $\beta = 7$ ,  $\gamma = 0$ ,  $\mu = 500$  for denoising 3D medical data of  $512 \times 512 \times 48$  voxels. Computational performance (CPU time in seconds / the number of iterations) for different numbers of subdomains ( $D = 1, 2, 4, 8$ ). The algorithms are stopped as soon as  $\|\mathbf{p}^{(k)} - \mathbf{p}^{(k-1)}\| < 10^{-6}$ .

that since no neighboring 2D-slices are taken into account, high variations in the vertical direction, i.e., the direction perpendicular to the 2D-slices, are allowed; see Figure 4.6 right column. A comparison of the solutions of this suboptimal strategy with our proposed domain decomposition algorithm applied to the genuine 3D image restoration problem shows that edges as well as uniform parts are much better preserved in the vertical direction, when dealing with 3D data rather than stacking 2D reconstructions; compare Figure 4.6. Hence, our proposed method is superior to the strategy which solves the problem slice by slice. This also illustrates that a careful treatment of the interfaces of domain patches is crucial.

Moreover, we compare the performance of Algorithm 1 with its parallel version, i.e., Algorithm 2, and with [25, Algorithm B] for denoising the 3D medical data of  $512 \times 512 \times 48$  voxels, considered already above. For the parameters we choose  $\alpha = 0.013$ ,  $\beta = 7$ ,  $\gamma = 0$ , and  $\mu = 500$  for different numbers of subdomains ( $D = 1, 2, 4, 8$ ). We terminate the algorithms as soon as the norm of the difference of two successive iterates drops below a certain threshold. More precisely, we use as a stopping criterion  $\|\mathbf{p}^{(k)} - \mathbf{p}^{(k+1)}\| < 10^{-6}$ , which seems suitable for our purposes. In fact, if our algorithm converges at least linearly, i.e., there exists an  $\varepsilon \in (0, 1)$  and an  $m > 0$  such that for all  $k \geq m$  we have  $\|\mathbf{p}^{(k)} - \mathbf{p}^{(k+1)}\| \leq \varepsilon \|\mathbf{p}^{(k)} - \mathbf{p}^{(\infty)}\|$ , the above stopping criterion ensures that the distance between our obtained result  $\mathbf{p}$  and  $\mathbf{p}^{(\infty)}$  is  $\|\mathbf{p} - \mathbf{p}^{(\infty)}\| \leq \frac{10^{-6}\varepsilon}{1-\varepsilon}$ . Note that in the parallel domain decomposition algorithm for the update of the iterate  $\mathbf{p}^{(k+1)}$  an averaging between the previous iterate  $\mathbf{p}^{(k)}$  and the current iterates  $\mathbf{p}_i^{(k+1)}$ ,  $i = 1 \dots, D$ , has to be performed; cf. Algorithm 2. Hence, we expect that the parallel algorithm needs more iterations until successful termination. Additionally, the communication time of the processors adds to the overall computing time. Nevertheless it turns out that the parallel method converges faster in time than the sequential algorithm for a sufficiently large number of subdomains; see Table 4.7. In fact, the parallel method with a decomposition into 8 domains performs best. These results confirm our previous observation in Section 4.5.1, where we already anticipated that the parallel version is more efficient than the sequential method for large image data sizes, as it is the case for 3D image data sets. Moreover, we observe that for such large data sets, the parallel as well as the sequential domain decomposition algorithms outperform the algorithm without domain splitting.

#### 4. Domain Decomposition for Dual Total Variation



**Figure 4.6:** Reconstruction of 3D CT data given by 48 slices. In the left column we show the reconstruction by the domain decomposition algorithm by splitting the data into 3D blocks, while the right column shows the reconstruction of a slice by slice reconstruction without taking the neighbor slices into account, i.e., solving 2D problems. In (c) and (d) we zoomed in on the in (a) and (b) highlighted area. In order to visualize the differences in the reconstruction of the two approaches we color in (e) and (f) the zoomed area.

## I. Subspace Correction Methods for Total Variation Minimization

### 4.6. References

- [1] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, 2000.
- [2] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing*, volume 147 of *Applied Mathematical Sciences*. Springer, New York, second edition, 2006.
- [3] D. P. Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.
- [4] C. Carstensen. Domain decomposition for a non-smooth convex minimization problem and its application to plasticity. *Numerical Linear Algebra with Applications*, 4(3):177–190, 1997.
- [5] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging Vision*, 20(1-2):89–97, 2004.
- [6] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock. An introduction to total variation for image analysis. *Theoretical foundations and numerical methods for sparse recovery*, 9:263–340, 2010.
- [7] A. Chambolle and J. Darbon. On total variation minimization and surface evolution using parametric maximum flows. *International Journal of Computer Vision*, 84(3):288–307, 2009.
- [8] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, 1997.
- [9] T. F. Chan, G. H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.
- [10] T. F. Chan and T. P. Mathew. Domain decomposition algorithms. *Acta Numerica*, 3:61–143, 1994.
- [11] T. F. Chan and J. J. Shen. *Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods*. SIAM, 2005.
- [12] H. Chang, X. Zhang, X.-C. Tai, and D. Yang. Domain decomposition methods for nonlocal total variation image restoration. *Journal of Scientific Computing*, 60(1):79–100, 2014.
- [13] K. Chen and X.-C. Tai. A nonlinear multigrid method for total variation minimization from image restoration. *Journal of Scientific Computing*, 33(2):115–138, 2007.
- [14] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.
- [15] I. Daubechies, G. Teschke, and L. Vese. Iteratively solving linear inverse problems under general convex constraints. *Inverse Problems and Imaging*, 1(1):29–46, 2007.
- [16] D. C. Dobson and C. R. Vogel. Convergence of an iterative method for total variation denoising. *SIAM Journal on Numerical Analysis*, 34(5):1779–1791, 1997.
- [17] Y. Duan and X.-C. Tai. Domain decomposition methods with graph cuts algorithms for total variation minimization. *Advances in Computational Mathematics*, 36(2):175–199, 2012.
- [18] L. C. Evans and R. F. Gariepy. *Measure theory and fine properties of functions*. Studies in Advanced Mathematics. CRC Press, Boca Raton, FL, 1992.
- [19] M. Fornasier, Y. Kim, A. Langer, and C.-B. Schönlieb. Wavelet decomposition method for  $L_2$ /TV-image deblurring. *SIAM Journal on Imaging Sciences*, 5(3):857–885, 2012.
- [20] M. Fornasier, A. Langer, and C.-B. Schönlieb. Domain decomposition methods for compressed sensing. In *Proceedings of the International Conference of SampTA09, Marseilles*, 2009.

#### 4. Domain Decomposition for Dual Total Variation

- [21] M. Fornasier, A. Langer, and C.-B. Schönlieb. A convergent overlapping domain decomposition method for total variation minimization. *Numerische Mathematik*, 116(4):645–685, 2010.
- [22] M. Fornasier and C.-B. Schönlieb. Subspace correction methods for total variation and  $l_1$ -minimization. *SIAM Journal on Numerical Analysis.*, 47(5):3397–3428, 2009.
- [23] G. Gilboa and S. Osher. Nonlocal operators with applications to image processing. *Multiscale Modeling & Simulation*, 7(3):1005–1028, 2008.
- [24] E. Giusti. *Minimal Surfaces and Functions of Bounded Variation*, volume 80 of *Monographs in Mathematics*. Birkhäuser Verlag, Basel, 1984.
- [25] M. Hintermüller and K. Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM Journal on Applied Mathematics*, 64(4):1311–1333, 2004.
- [26] M. Hintermüller and A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1/L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.
- [27] M. Hintermüller and A. Langer. Surrogate functional based subspace correction methods for image processing. In *Domain Decomposition Methods in Science and Engineering XXI*, pages 829–837. Springer, 2014.
- [28] M. Hintermüller and C. Rautenberg. On the density of classes of closed convex sets with pointwise constraints in Sobolev spaces. *Journal of Mathematical Analysis and Applications*, 426(1):585–593, 2015.
- [29] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.
- [30] K. Ito and K. Kunisch. *Lagrange multiplier approach to variational problems and applications*, volume 15 of *Advances in Design and Control*. SIAM, Philadelphia, PA, 2008.
- [31] A. Langer, S. Osher, and C.-B. Schönlieb. Bregmanized domain decomposition for image restoration. *Journal of Scientific Computing*, 54(2-3):549–576, 2013.
- [32] J. Müller, M. Burger, and S. Gorlatch. Parallel total variation minimization. *University of Münster, Diploma Thesis*, 2008.
- [33] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming Series A*, 103(1):127–152, 2005.
- [34] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.
- [35] G. Peyré, S. Bougleux, and L. Cohen. Non-local regularization of inverse problems. *Inverse Problems and Imaging*, 5(2):511–530, 2011.
- [36] R. T. Rockafellar. *Convex Analysis*. Princeton Mathematical Series, No. 28. Princeton University Press, Princeton, N.J., 1970.
- [37] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.
- [38] H. A. Schwarz. Über einige Abbildungsaufgaben. *Journal für Reine Angewandte Mathematik*, 70:105–120, 1869.
- [39] X.-C. Tai. Rate of convergence for some constraint decomposition methods for nonlinear variational inequalities. *Numerische Mathematik*, 93(4):755–786, 2003.
- [40] X.-C. Tai and P. Tseng. Convergence rate analysis of an asynchronous space decomposition method for convex minimization. *Mathematics of Computation*, 71(239):1105–1135, 2002.

## I. Subspace Correction Methods for Total Variation Minimization

- [41] X.-C. Tai and J. Xu. Global and uniform convergence of subspace correction methods for some convex optimization problems. *Mathematics of Computation*, 71(237):105–124, 2002.
- [42] P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117(1):387–423, 2009.
- [43] L. Vese. A study in the BV space of a denoising-deblurring variational problem. *Applied Mathematics & Optimization*, 44(2):131–161, 2001.
- [44] J. Xu, X.-C. Tai, and L.-L. Wang. A two-level domain decomposition method for image restoration. *Inverse Problems and Imaging*, 4(3):523–545, 2010.
- [45] J. Xu and J. Zou. Some nonoverlapping domain decomposition methods. *SIAM Review*, 40(4):857–914, 1998.
- [46] X. Zhang, M. Burger, X. Bresson, and S. Osher. Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM Journal on Imaging Sciences*, 3(3):253–276, 2010.
- [47] X. Zhang and T. F. Chan. Wavelet inpainting by nonlocal total variation. *Inverse Problems and Imaging*, 4(1):191–210, 2010.

# 5. Overlapping Domain Decomposition Methods for Total Variation Denoising

**Bibliographic Note:** The content of this chapter contains the results of the following recently submitted manuscript [AL4]: A. Langer. Overlapping domain decomposition methods for total variation denoising. *Preprint, submitted to SIAM Journal on Numerical Analysis*, 2018.

**Summary:** Alternating and parallel overlapping domain decomposition methods for the minimization of the total variation are presented. Their derivation is based on the predual formulation of the total variation minimization problem. In particular, the predual total variation minimization problem is decomposed into overlapping domains yielding subdomain problems in the respective dual space. Subsequently these subdomain problems are again dualized, forming a splitting algorithm for the original total variation minimization problem. The convergence of the proposed domain decomposition methods to a solution of the global problem is proven. In contrast to other works, the analysis is carried out in an infinite dimensional setting. Numerical experiments are shown to support the theoretical results and to demonstrate the effectiveness of the algorithms.

## 5.1. Introduction

Minimizing the total variation in the context of image denoising was first proposed in [49] and has gained a lot of attention since then, because it allows to preserve edges and discontinuities in images. In this approach one typically minimizes an energy consisting of a data-fidelity term, which enforces the consistency between the observed and obtained image, a total variation term, as regularizer, and a parameter weighting the importance of these two terms. The choice of the data-term depends on the type of noise contamination. Here we assume that the observed image  $g$  is corrupted by additive Gaussian noise, i.e.,  $g = \hat{u} + \eta$ , where  $\hat{u}$  is the unknown true image and  $\eta$  represents the noise. For such images usually a quadratic  $L^2$ -data fidelity term is chosen; see for example [6, 9] and references therein. That is, the image  $\hat{u}$  is recovered from the observation  $g \in L^2(\Omega)$  by solving

$$\arg \min_{u \in L^2(\Omega)} \left\{ J(u) := \frac{1}{2} \|u - g\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} |Du| \right\} \quad (5.1)$$

where  $\Omega \subset \mathbb{R}^2$  is an open bounded set with Lipschitz boundary,  $\alpha > 0$  is the regularization parameter and  $\int_{\Omega} |Du| = \sup \{ \int_{\Omega} u \operatorname{div} \mathbf{p} dx : \mathbf{p} \in C_0^1(\Omega, \mathbb{R}^2), |\mathbf{p}|_{\ell^2} \leq 1 \text{ almost everywhere (a.e.)} \}$  denotes the total variation of  $u$  in  $\Omega$  [3, 29] with  $C_0^1(\Omega, \mathbb{R}^2)$  being the space of continuously differentiable vector valued functions with compact support in  $\Omega$ . Here and in the rest of the paper, bold letters indicate vector valued functions. Note, that other and different fidelity terms have been considered in connection with other types of noise, as impulsive

## I. Subspace Correction Methods for Total Variation Minimization

noise [2, 46, 47], Poisson noise [44], multiplicative noise [4], Rician noise [28], mixed noise [7, 15, 19, 32, 40, 42].

Existing state-of-the-art methods for solving (5.1), as described in [6, 9], perform well for small- and medium-scale problems. However, they are not able to perform in realistic CPU-time large-scale problems. Such large-scale problems, occur nowadays in nearly every application in image reconstruction, due to the improvement of hardware. In order to deal with these huge problems, new methods need to be developed.

It has been shown multiple times [48, 53], that domain decomposition methods are one of the most successful methods to construct efficient solvers for large-scale problems. This is due to the fact, that they allow for decomposing the original problem into a sequence of smaller problems, which may be distributed on several processors with the possibility of parallelization. While for domain decomposition methods minimizing smooth energies, the convergence, rate of convergence, and independence of the rate of convergence from the mesh size of discretization are well established, not much is known for minimizing nonsmooth and nonadditive functionals. Note, that the energy in (5.1) is nonsmooth and nonadditive, due to the presence of the total variation. We remark, that for nonsmooth problems, the resulting splitting algorithms still work fine as long as the energy splits additively with respect to the domain decomposition. For such problems convergence and sometimes even rate of convergence are ensured; see for example [23, 54]. Moreover, for image deblurring problems preconditioning effects of a specific subspace correction algorithm for minimizing a nonsmooth energy are shown in [55]. For nonsmooth and nonadditive energies, however, the research on subspace correction methods is far from being complete, and for some problem classes counterexamples do exist indicating failure of splitting techniques; see e.g. [24, 57].

For introducing domain decomposition strategies for solving problem (5.1) the major difficulty lies in the correct treatment of the interfaces of the domain decomposition patches, i.e. the preservation of crossing discontinuities and the correct matching where the solution is continuous. We emphasize that for well-known approaches as those in [8, 11, 51, 52] it is not clear yet whether they indeed converge to a global minimizer for nonsmooth and nonadditive problems, as any convergence theory in this direction is missing. Nevertheless, in [14] and [58] the subspace correction approaches of [51, 52] are used to solve smoothed versions of (5.1).

In [25, 26, 27, 43, 50] nonoverlapping and overlapping domain decomposition methods for total variation minimization are described. Thereby, the convex objective under some linear constraint, ensuring the correct treatment of the internal interfaces, is iteratively minimized on each subdomain. While in these papers an implementation guaranteeing convergence and monotonic decay of the objective energy is provided, convergence to the global minimizer of the total variation problem cannot be ensured, in general. In [26] a proof, establishing convergence of overlapping domain decomposition algorithms to the global solution in a discrete setting, is presented, which, however, only holds for one-dimensional problems. It is not clear yet how to extend this proof to any finite dimensional space without introducing additional assumptions. Moreover, an extension to infinite dimensional spaces is also missing till now.

For a class of nonsmooth and nonadditive convex variational problems with a combined  $L^1/L^2$  data fidelity term in [32, 33] overlapping and nonoverlapping domain decomposition methods are presented. In particular, their convergence and monotonic decay of the energy is theoretically ensured. Moreover, an estimate of the distance of the limit point obtained from the domain decomposition methods to the true global minimizer is derived. With the help of this estimate it is demonstrated by numerical experiments that the domain decomposition methods indeed generate sequences which converge to the global minimizer. However, a the-

## 5. Overlapping Domain Decomposition for Total Variation Denoising

oretical proof of convergence of the domain decomposition methods to the global solution is missing.

Without any rigorous theoretical analysis in [21] domain decomposition methods for solving (5.1) by graph cuts are introduced and applied to the task of image segmentation. Moreover, for image segmentation using the Chan-Vese model [12] and based on a primal-dual formulation recently nonoverlapping domain decomposition methods are presented in [20].

In order to tackle the difficulties due to the minimization of a nonsmooth and nonadditive objective in (5.1), in [13, 34] a predual problem of (5.1), see [31, 38] for the derivation of the latter, is considered. In fact, a predual of (5.1) reads:

$$\begin{aligned} & \min \frac{1}{2} \|\operatorname{div} \mathbf{p} + g\|_{L^2(\Omega)}^2 \quad \text{over } \mathbf{p} \in H_0(\operatorname{div}, \Omega) \\ & \text{s.t. } |\mathbf{p}(x)|_{\ell^2} \leq \alpha \text{ for almost all (f.a.a.) } x \in \Omega, \end{aligned} \quad (5.2)$$

where  $H_0(\operatorname{div}, \Omega) := \{\mathbf{v} \in \mathbb{L}^2(\Omega) : \operatorname{div} \mathbf{v} \in L^2(\Omega), \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega\}$  with  $\mathbb{L}^2(\Omega) := L^2(\Omega) \times L^2(\Omega)$  and  $\mathbf{n}$  being the outward unit normal on  $\partial\Omega$ . Note, that the solution  $u^*$  of (5.1) and a solution  $\mathbf{p}^*$  of (5.2) are related by

$$u^* = \operatorname{div} \mathbf{p}^* + g, \quad (5.3)$$

see [31]. The smooth objective and the box-constraint seem more amenable to a domain decomposition than the structure of (5.1). In fact, in [34] nonoverlapping domain decomposition methods for the problem in (5.2), where instead of  $|\mathbf{p}(x)|_{\ell^2} \leq \alpha$  the constraint  $|\mathbf{p}(x)|_{\ell^\infty} \leq \alpha$  is considered, are introduced and shown to converge to a minimizer of the global problem in a discrete setting. It is still an open problem to show such a convergence result in an infinite dimensional setting.

Based on the nonoverlapping domain decomposition strategy in [34] for the predual problem (5.2) in [45] a nonoverlapping algorithm for a discretized version of the primal problem (5.1) is constructed. Thereby the following strategy is pursued: The domain decomposition method in [34] is constituted by its subdomain problems. Then the dual problems of these subdomain problems are computed, yielding a sequence of nonoverlapping subdomain problems of the primal problem. Due to the predualization and dualization, the finally constituted domain decomposition method of the discretized primal problem looks different than the splitting strategies presented in [27, 32]. Using the connection between the primal subdomain problems and their predual counterparts allows in [45] to show analytically the convergence of the nonoverlapping domain decomposition methods to the minimizer of the global problem in a discrete setting.

While convergence to the solution of the global problems (5.1) or (5.2) for nonoverlapping domain decomposition methods in an infinite dimensional setting has not yet been theoretically proven, there exist overlapping domain decomposition methods for the predual problem (5.2) with such desirable convergence property. In particular, in [13] overlapping domain decomposition methods for the predual problem (5.2) are introduced and the convergence to the true minimizer of the global problem is shown analytically in an infinite dimensional setting, even together with a convergence rate. We note, that this results cannot be (directly) extended to the nonoverlapping case. This is due to the fact, that the overlapping decomposition in [13] is determined by a partition of unity function, whose partitions have to be sufficiently smooth. This essential smoothness property is unfortunately lost for a nonoverlapping splitting.

We summarize, that domain decomposition algorithms with a theoretical guarantee to convergence to the minimizer of the global problem are till now given for (i) the discrete predual

## I. Subspace Correction Methods for Total Variation Minimization

problem with a nonoverlapping decomposition [34], (ii) the continuous predual problem with an overlapping decomposition [13], and (iii) the discrete primal problem with a nonoverlapping decomposition [45].

In this paper, we continue making up this list by presenting convergent sequential and parallel overlapping domain decomposition methods for the primal problem (5.1). In particular we prove their convergence to the solution of (5.1) in an infinite dimensional setting. Thereby we follow basically the idea of [45]. That is, we consider the overlapping domain decomposition method in [13], which is stated in a continuous setting, and compute from its subdomain problems their respective dual problems. This yields an overlapping domain decomposition method for the problem in (5.1). Again, due to the connection of the subdomain problems to the predual subdomain problems of the method in [13], we are able to prove that our newly proposed methods converge to a minimizer of the underlying global problem. Although the proofs presented in this paper are motivated by the ones in [45], our analysis differs significantly from the one in [45]. This is due to the following reasons: Firstly, we consider overlapping domain decomposition methods, while in [45] nonoverlapping methods are considered. In particular, our subdomain problems are very different to the ones in [45], asking for different subdomain solvers as well as for a different convergence analysis. Secondly, while the analysis in [45] is carried out in a discrete setting, our analysis as well as our methods are presented in a continuous setting, which generates additional difficulties in proving convergence. For example, while in a discrete setting bounded sequences have strongly converging subsequences, a bounded sequence in  $L^2$  has (only) weakly converging subsequences. Moreover, the gradient of a bounded function is in general not bounded in  $L^2$ , while in a discrete setting the gradient of any bounded function is bounded again.

We found that in a discrete setting our proposed overlapping strategy allows for overlapping regions of size zero, yielding a nonoverlapping decomposition. In particular, in this limit case our proposed splitting strategies become the methods in [45], see Remark 5.18 below. Hence, in this sense our algorithms generalize the ones in [45].

The rest of the paper is organized as follows: In 5.2 we derive and present the proposed alternating and parallel domain decomposition methods. Their convergence to a minimizer of the global problem (5.1) is shown analytically. In Section 5.3 we describe two different approaches on how to solve the subspace minimization problems. In particular, one approach first restricts the subspace problems to the subdomain and then discretizes the problem accordingly, while the other approach first discretizes the subproblems and then restricts it to the respective subdomain. Numerical experiments for the alternating and parallel domain decomposition methods are shown in Section 5.4, demonstrating the efficiency of the methods.

## 5.2. Overlapping Domain Decomposition Algorithms

In this section we present and analyze the proposed overlapping domain decomposition algorithms. Their derivation is based on the splitting strategy in [13]. However we start by fixing some notations.

### 5.2.1. Basic Terminology

For a Banach space  $V$  we denote by  $V'$  its topological dual and  $\langle \cdot, \cdot \rangle_{V' \times V}$  describes the bilinear canonical pairing over  $V' \times V$ . The norm of a Banach space  $V$  is written as  $\|\cdot\|_V$ . For any

## 5. Overlapping Domain Decomposition for Total Variation Denoising

$\mathbf{p} = (p^1, p^2) \in \mathbb{L}^s(\Omega)$ ,  $1 \leq s \leq \infty$ , we define the norm  $\|\mathbf{p}\|_{\mathbb{L}^s(\Omega)} = (\|p^1\|_{L^s(\Omega)}^2 + \|p^2\|_{L^s(\Omega)}^2)^{1/2}$ . By  $(\cdot, \cdot)$  we denote the standard inner product in  $L^2(\Omega)$ .

For ease of notation, in the sequel for any sequence  $(v^n)_{n \in \mathbb{N}}$  we write  $(v^n)_n$  instead. A convex functional  $\mathcal{F} : V \rightarrow \overline{\mathbb{R}} := \mathbb{R} \cup \{+\infty\}$  is called *proper*, if  $\{v \in V : \mathcal{F}(v) \neq +\infty\} \neq \emptyset$  and  $\mathcal{F}(v) > -\infty$  for all  $v \in V$ . A functional  $\mathcal{F} : V \rightarrow \overline{\mathbb{R}}$  is called *(weakly) lower semicontinuous* (*l.s.c.*), if for every (weakly) convergent sequence  $(v^n)_n \subset V$  with limit  $v \in V$  we have

$$\liminf_{n \rightarrow \infty} \mathcal{F}(v^n) \geq \mathcal{F}(v).$$

Note, that in infinite dimensional spaces, weak l.s.c. is a stronger requirement than (strong) l.s.c. However, a convex and (strongly) l.s.c. function is weakly l.s.c. thanks to Mazur's lemma. Further, weak l.s.c. and (strong) l.s.c. coincide in finite dimensional spaces [5]. In the context of convergence we often use the symbols “ $\rightarrow$ ” and “ $\rightharpoonup$ ” to indicate strong and weak convergence respectively.

For a convex functional  $\mathcal{F} : V \rightarrow \overline{\mathbb{R}}$  we define the *subdifferential* of  $\mathcal{F}$  at  $v \in V$  as the set valued function

$$\partial\mathcal{F}(v) := \begin{cases} \emptyset & \text{if } \mathcal{F}(v) = \infty, \\ \{v^* \in V' : \langle v^*, u - v \rangle_{V' \times V} + \mathcal{F}(v) \leq \mathcal{F}(u) \quad \forall u \in V\} & \text{otherwise.} \end{cases}$$

It is clear from this definition, that  $0 \in \partial\mathcal{F}(v)$  if and only if  $v$  is a minimizer of  $\mathcal{F}$ .

The *conjugate function* (or *Legendre transform*) of a convex function  $\mathcal{F} : V \rightarrow \overline{\mathbb{R}}$  is defined as  $\mathcal{F}^* : V' \rightarrow \overline{\mathbb{R}}$  with

$$\mathcal{F}^*(v^*) = \sup_{v \in V} \{\langle v^*, v \rangle_{V' \times V} - \mathcal{F}(v)\}.$$

From this definition we see that  $\mathcal{F}^*$  is the pointwise supremum of continuous affine functions and thus, according to [22, Proposition 3.1, p 14], convex, lower semicontinuous, and proper.

A functional  $\mathcal{F} : V \rightarrow \overline{\mathbb{R}}$  is said to be *coercive* (in  $V$ ), if for every sequence  $(v^n)_n \subset V$  with  $\|v^n\|_V \rightarrow \infty$ , we have  $\mathcal{F}(v^n) \rightarrow \infty$ . Let  $V, W$  be two Banach spaces, then for any operator  $\Lambda : V \rightarrow W$  we define by  $\Lambda^* : W' \rightarrow V'$  its *adjoint*.

In the sequel we will often use  $C(\Omega)$ , the space of continuous functions in  $\Omega$ ,  $C_0(\Omega, \mathbb{R}^2)$ , the space of  $\mathbb{R}^2$ -valued continuous functions with compact support in  $\Omega$ , and  $\mathcal{D}(\Omega)$ , the space of infinitely differentiable functions with compact support in  $\Omega$ . By  $H^1(\Omega)$  we denote the Sobolev space  $W^{1,2}(\Omega)$ , i.e., the space of functions in  $L^2(\Omega)$ , whose first weak derivatives are again in  $L^2(\Omega)$ .

### 5.2.2. Preliminaries

For ease of presentation, and in order to avoid unnecessary technicalities, we limit the derivation of our proposed domain decomposition algorithms to a splitting into 2 subdomains, by noting, that the generalization to multiple domains comes quite natural, see Section 5.2.5. Hence, we are considering an overlapping decomposition of the image domain  $\Omega$  into 2 subdomains  $\Omega_1$  and  $\Omega_2$ , such that  $\Omega = \Omega_1 \cup \Omega_2$  and  $\Omega_1 \cap \Omega_2 \neq \emptyset$ . In order to compute a minimizer of (5.1) with respect to such a splitting, in [13] based on the predual formulation (5.2) the algorithm shown in Algorithm 1 is proposed.

In Algorithm 1  $(\theta_i)_{i=1,2}$  denotes a partition of unity function with the properties

- (a)  $\theta_1 + \theta_2 \equiv 1$  and  $\theta_i \geq 0$  a.e. on  $\Omega$ ,

## I. Subspace Correction Methods for Total Variation Minimization

---

**Algorithm 1** Alternating predual version from [13]

---

```

Initialize:  $\mathbf{p}^0 \in H_0(\text{div}, \Omega)$  and  $\hat{\alpha} \in (0, 1]$ 
for  $n = 0, 1, 2, \dots$  do
     $\hat{\mathbf{q}}_1^n \in \arg \min \left\{ \frac{1}{2} \| \text{div}(\mathbf{v} + \theta_2 \mathbf{p}^n) + g \|_{L^2(\Omega)}^2 : \mathbf{v} \in H_0(\text{div}, \Omega), |\mathbf{v}(x)|_{\ell^2} \leq \alpha \theta_1(x) \text{ f.a.a. } x \in \Omega \right\}$ 
     $\mathbf{q}_1^n = (1 - \hat{\alpha}) \theta_1 \mathbf{p}^n + \hat{\alpha} \hat{\mathbf{q}}_1^n$ 
     $\hat{\mathbf{q}}_2^n \in \arg \min \left\{ \frac{1}{2} \| \text{div}(\mathbf{v} + \mathbf{q}_1^n) + g \|_{L^2(\Omega)}^2 : \mathbf{v} \in H_0(\text{div}, \Omega), |\mathbf{v}(x)|_{\ell^2} \leq \alpha \theta_2(x) \text{ f.a.a. } x \in \Omega \right\}$ 
     $\mathbf{q}_2^n = (1 - \hat{\alpha}) \theta_2 \mathbf{p}^n + \hat{\alpha} \hat{\mathbf{q}}_2^n$ 
     $\mathbf{p}^{n+1} = (1 - \hat{\alpha}) \mathbf{p}^n + \hat{\alpha} (\hat{\mathbf{q}}_1^n + \hat{\mathbf{q}}_2^n)$ 
end for

```

---

- (b)  $\text{supp } \theta_i \subset \bar{\Omega}_i$ ,
- (c)  $\theta_i \in H^1(\Omega)$ ,  $\|\nabla \theta_i\|_{L^\infty(\Omega)} \leq c_0$ ,

for  $i = 1, 2$ , where  $c_0 > 0$  is a constant depending on the overlapping-size; see [13]. Based on the relation (5.3) in [13] it is shown, that Algorithm 1 generates a sequence  $(\mathbf{p}^n)_n$  such that  $(u^n)_n$ , where  $u^n = \text{div } \mathbf{p}^n + g$ , converges to a minimizer of (5.1).

For deriving our algorithms, following the idea of [45], we calculate the dual problems of the subdomain problems of Algorithm 1. The subdomain problems in Algorithm 1 may be rewritten as

$$\arg \min \left\{ \frac{1}{2} \| \text{div } \mathbf{v} + f \|_{L^2(\Omega)}^2 : \mathbf{v} \in H_0(\text{div}, \Omega), |\mathbf{v}(x)|_{\ell^2} \leq \lambda(x) \text{ f.a.a. } x \in \Omega \right\} \quad (5.4)$$

where  $f = \text{div } \theta_2 \mathbf{p}^n + g$ ,  $\lambda = \alpha \theta_1$  in  $\Omega_1$  and  $f = \text{div } \mathbf{q}_1^n + g$ ,  $\lambda = \alpha \theta_2$  in  $\Omega_2$  for any  $n \geq 0$ . Note that, since  $f \in L^2(\Omega)$  in both situations and  $\lambda : \bar{\Omega} \rightarrow \mathbb{R}_0^+$  is a bounded function, the existence of a minimizer of (5.4) is ensured [36, Proposition 3.2 (b)]. If  $\lambda \in C(\bar{\Omega})$  and  $\lambda(x) > 0$  for all  $x \in \bar{\Omega}$ , then a dual problem of (5.4) is given by

$$\arg \min_{u \in L^2(\Omega)} \left\{ \tilde{J}(u) := \frac{1}{2} \|u - f\|_{L^2(\Omega)}^2 + \int_{\Omega} \lambda |Du| \right\}, \quad (5.5)$$

and has a unique solution, see [36, 37]. Here and in the sequel, the expression  $\int_{\Omega} \lambda |Du|$  describes the integral of  $\lambda$  on  $\Omega$  with respect to the measure  $|Du|$ , where  $Du$  is the distributional gradient of  $u$ , see [36] for more details. In this situation, thanks to [31, 36], we have the following relation between (5.4) and (5.5):

**Lemma 5.1.** *Let  $f \in L^2(\Omega)$  be given and  $\lambda : \bar{\Omega} \rightarrow \mathbb{R}^+$  with  $\lambda \in C(\bar{\Omega})$  (i.e.,  $\lambda$  is a positive continuous bounded function). Then  $u^*$  is a minimizer of (5.5) if and only if there exists a  $\mathbf{p}^* \in H_0(\text{div}, \Omega)$  such that*

- (i)  $\text{div } \mathbf{p}^* + f = u^*$  and
- (ii)  $\mathbf{p}^* \in \arg \min_{\mathbf{p} \in K} \frac{1}{2} \| \text{div } \mathbf{p} + f \|_{L^2(\Omega)}^2$  where  $K = K(\Omega) := \{ \mathbf{p} \in H_0(\text{div}, \Omega) : |\mathbf{p}(x)|_{\ell^2} \leq \lambda(x) \text{ f.a.a. } x \in \Omega \}$ .

Further,  $\mathbf{p}^* \in \arg \min_{\mathbf{p} \in K} \frac{1}{2} \| \text{div } \mathbf{p} + f \|_{L^2(\Omega)}^2$  if and only if

$$\langle (-\text{div})^*(\text{div } \mathbf{p}^* + f), \mathbf{p} - \mathbf{p}^* \rangle_{H_0(\text{div}, \Omega)' \times H_0(\text{div}, \Omega)} \leq 0 \quad \forall \mathbf{p} \in K$$

and  $\mathbf{p}^* \in K$ .

## 5. Overlapping Domain Decomposition for Total Variation Denoising

In our case  $\lambda$  is not necessarily positive, i.e.,  $\lambda : \bar{\Omega} \rightarrow \mathbb{R}_0^+$ , since it is in (5.5) the product of the regularization parameter  $\alpha$  and a partition of unity function, i.e.,  $\lambda = \alpha\theta_i$  for  $i \in \{1, 2\}$ . This more general case is not covered in [36]. In particular, while in [36] the existence of minimizers of (5.4) with nonnegative, continuous and bounded  $\lambda$  is shown, no attention is given to problem (5.5) with such functions  $\lambda$ . Therefore, we show next the existence of a minimizer for (5.5) with  $\lambda : \bar{\Omega} \rightarrow \mathbb{R}_0^+$  being continuous, i.e.,  $\lambda$  is nonnegative, continuous and bounded. In the sequel we assume, that  $\text{supp}(\lambda) \subseteq \bar{\Omega}$  and  $\Omega_0 := \Omega \setminus \text{supp}(\lambda)$  is then open. That is, if  $\text{supp } \lambda = \bar{\Omega}$ , then  $\Omega_0 = \emptyset$ , otherwise, i.e.,  $\text{supp } \lambda \subsetneq \bar{\Omega}$ , then  $\Omega_0 \neq \emptyset$ . Since  $\Omega_0 = \text{int}(\Omega_0)$ , it follows that if  $\mathbf{q} \in C_0^1(\Omega, \mathbb{R}^2)$  and constant in  $\Omega_0$ , i.e.,  $\mathbf{q}(x) = \mathbf{c} \in \mathbb{R}^2$  f.a.a.  $x \in \Omega_0$ , then  $\text{div } \mathbf{q} = 0$  a.e. on  $\Omega_0$ .

**Lemma 5.2.** *Let  $u \in L^2(\Omega)$ ,  $\lambda : \bar{\Omega} \rightarrow \mathbb{R}_0^+$ ,  $\lambda \in H^1(\Omega) \cap C(\bar{\Omega})$ ,  $\|\nabla \lambda\|_{L^\infty(\Omega)} < \infty$ , and  $\text{supp}(\lambda) \subseteq \bar{\Omega}$ , then*

$$\sup_{\mathbf{p} \in K(\Omega)} (u, -\text{div } \mathbf{p}) = \int_{\Omega} \lambda |Du| = \int_{\text{supp}(\lambda)} \lambda |Du|.$$

*Proof.* We get

$$\begin{aligned} \int_{\text{supp}(\lambda)} \lambda |Du| &= \int_{\Omega \setminus \Omega_0} \lambda |Du| = \sup_{\mathbf{p} \in \mathcal{K}(1, C_0(\Omega \setminus \Omega_0, \mathbb{R}^2))} \langle \lambda Du, \mathbf{p} \rangle_{C_0(\Omega \setminus \Omega_0, \mathbb{R}^2)' \times C_0(\Omega \setminus \Omega_0, \mathbb{R}^2)} \\ &= \sup_{\mathbf{p} \in \mathcal{K}(1, C_0(\Omega, \mathbb{R}^2))} \langle \lambda Du, \mathbf{p} \rangle_{C_0(\Omega, \mathbb{R}^2)' \times C_0(\Omega, \mathbb{R}^2)} = \int_{\Omega} \lambda |Du|, \end{aligned}$$

since  $\lambda \in C(\bar{\Omega})$  and  $\lambda(x) = 0$  f.a.a.  $x \in \Omega_0$ , where  $\mathcal{K}(\lambda, C_0(\Omega, \mathbb{R}^2)) := \{\mathbf{p} \in C_0(\Omega, \mathbb{R}^2) : |\mathbf{p}(x)|_{\ell^2} \leq \lambda(x) \text{ f.a.a. } x \in \Omega\}$ . Since  $\mathcal{K}(1, \mathcal{D}(\Omega)^2)$  is dense in  $\mathcal{K}(1, C_0(\Omega, \mathbb{R}^2))$  in the sense of  $C_0(\Omega, \mathbb{R}^2)$ , and  $\mathcal{K}(1, \mathcal{D}(\Omega)^2)$  is dense in  $\mathcal{K}(1, H_0(\text{div}, \Omega))$  in the sense of  $H_0(\text{div}, \Omega)$  [35], we observe  $\sup_{\mathbf{p} \in K(\Omega)} (u, -\text{div } \mathbf{p}) = \int_{\Omega} \lambda |Du|$ .  $\square$

Now we prove the existence and uniqueness of the minimizer of (5.5).

**Proposition 5.3.** *If  $\lambda : \bar{\Omega} \rightarrow \mathbb{R}_0^+$ ,  $\lambda \in C(\bar{\Omega})$ , and  $\text{supp } \lambda \subseteq \bar{\Omega}$ , then (5.5) admits a unique solution.*

*Proof.* The functional  $\tilde{J}$  is obviously bounded from below by 0 and there exists a  $v \in L^2(\Omega)$  such that  $\tilde{J}(v) \in \mathbb{R}$ . Further,  $\tilde{J}$  is coercive in  $L^2(\Omega)$ , since  $\|u - f\|_{L^2(\Omega)}^2 \leq \tilde{J}(u)$  for all  $u \in L^2(\Omega)$ . The weak lower semicontinuity of  $\tilde{J}$  follows by the continuity of the map  $u \rightarrow \int_{\Omega} -\text{div } \mathbf{p} u \, dx$ , where  $\mathbf{p} \in K$ . Then the direct method in the calculation of variations yields the existence of a minimizer.

The uniqueness of the minimizer follows by the strict convexity of  $\tilde{J}$ .  $\square$

Note, that a minimizer of  $\tilde{J}$  not necessarily is an element of  $BV(\Omega)$ . This is easily seen, since a minimizer is allowed to have infinite variation in  $\Omega_0$ . In order to show the duality relation between (5.4) and (5.5) for  $\lambda \geq 0$ , we recall the Fenchel duality theorem; see, e.g., [22] for more details.

**Theorem 5.4** (Fenchel duality theorem). *Let  $V$  and  $W$  be two Banach spaces with topological duals  $V'$  and  $W'$ , respectively, and  $\Lambda : V \rightarrow W$  a bounded linear operator with adjoint  $\Lambda^* \in \mathcal{L}(W', V')$ . Further let  $\mathcal{F} : V \rightarrow \bar{\mathbb{R}}$ ,  $\mathcal{G} : W \rightarrow \bar{\mathbb{R}}$  be convex, lower semicontinuous, and*

## I. Subspace Correction Methods for Total Variation Minimization

proper functionals. Assume there exists  $q_0 \in V$  such that  $\mathcal{F}(q_0) < \infty$ ,  $\mathcal{G}(\Lambda q_0) < \infty$  and  $\mathcal{G}$  is continuous at  $\Lambda q_0$ . Then we have

$$\inf_{q \in V} \mathcal{F}(q) + \mathcal{G}(\Lambda q) = \sup_{v \in W'} -\mathcal{F}^*(\Lambda^* v) - \mathcal{G}^*(-v) \quad (5.6)$$

and the problem on the right hand side of (5.6) admits a solution  $\bar{v}$ . Moreover,  $\bar{v}$  and  $\bar{q}$  are solutions of the two optimization problems in (5.6), respectively, if and only if

$$\begin{aligned} \Lambda^* \bar{v} &\in \partial \mathcal{F}(\bar{q}), \\ -\bar{v} &\in \partial \mathcal{G}(\Lambda \bar{q}). \end{aligned} \quad (5.7)$$

Now we are able to state our duality result.

**Theorem 5.5.** Let  $\lambda$  be defined as in Lemma 5.2. Then a Fenchel dual of (5.4) is given by (5.5). Moreover, let  $u^*$  be a minimizer of (5.5) and  $\mathbf{p}^*$  a minimizer of (5.4) then

$$u^* = \operatorname{div} \mathbf{p}^* + f \quad \text{and} \quad \langle (-\operatorname{div})^* u^*, \mathbf{p} - \mathbf{p}^* \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} \leq 0 \quad \forall \mathbf{p} \in K. \quad (5.8)$$

*Proof.* We apply the Fenchel duality result (see Theorem 5.4) with  $V = H_0(\operatorname{div}, \Omega)$ ,  $W = L^2(\Omega)$ ,  $\Lambda = -\operatorname{div}$ ,  $\mathcal{F} : V \rightarrow \mathbb{R}$  given by  $\mathcal{F}(\mathbf{p}) = \mathbf{I}_K(\mathbf{p})$ , and  $\mathcal{G} : L^2(\Omega) \rightarrow \mathbb{R}$  given by  $\mathcal{G}(v) = \frac{1}{2} \|v - f\|_{L^2(\Omega)}^2$ , where

$$\mathbf{I}_K(\mathbf{p}) = \begin{cases} 0 & \text{if } \mathbf{p} \in K \\ \infty & \text{otherwise} \end{cases}.$$

The convex conjugate  $\mathcal{G}^* : L^2(\Omega) \rightarrow \mathbb{R}$  is then  $\mathcal{G}^*(v) = \frac{1}{2} \|v + f\|_{L^2(\Omega)}^2 - \frac{1}{2} \|f\|_{L^2(\Omega)}^2$ , cf. [31], and the convex conjugate  $\mathcal{F}^* : V' \rightarrow \mathbb{R}$  is given by  $\mathcal{F}^*(\mathbf{q}) = \sup_{\mathbf{p} \in K} \langle \mathbf{q}, \mathbf{p} \rangle_{V' \times V} - \mathcal{F}(\mathbf{p})$ . Thus

$$\mathcal{F}^*(\Lambda^* v) = \sup_{\mathbf{p} \in K} (v, -\operatorname{div} \mathbf{p}) = \int_{\Omega} \lambda |Dv|,$$

where we used Lemma 5.2. From (5.7) we find (5.8).  $\square$

In order, that (5.5) is well defined and that convergence to the true minimizer of the proposed domain decomposition algorithms is guaranteed, a partition of unity function needs to have the following properties

$$(a') \quad \theta_1 + \theta_2 \equiv 1 \text{ and } \theta_i \geq 0 \text{ a.e. on } \overline{\Omega} \text{ for } i = 1, 2, \quad (5.9)$$

$$(b') \quad \operatorname{supp}(\theta_i) \subset \overline{\Omega}_i \text{ for } i = 1, 2, \quad (5.10)$$

$$(c') \quad \theta_i \in H^1(\Omega) \cap C(\overline{\Omega}) \text{ and } \|\nabla \theta_i\|_{L^\infty(\Omega)} < \infty \text{ for } i = 1, 2. \quad (5.11)$$

In the sequel we will only use a partition of unity function with the properties (a'), (b'), and (c'), and denote it by  $(\theta_i)_i$ .

**Remark 5.6.** The theory presented in this paper can be generalized to other types of total variation, i.e.,

$$\int_{\Omega} |Du|_r = \sup \left\{ \int_{\Omega} u \operatorname{div} \mathbf{p} \, dx : \mathbf{p} \in C_0^1(\Omega, \mathbb{R}^2), |\mathbf{p}|_{\ell^{r^*}} \leq 1 \text{ a.e. in } \Omega \right\}$$

with  $1 \leq r < +\infty$ , leading to a similar analysis and similar results; cf. Remark 5.16 below. For instance, the predual problem of (5.5) with  $\int_{\Omega} |Du|_r$  is given by

$$\arg \min \left\{ \frac{1}{2} \|\operatorname{div} \mathbf{v} + f\|_{L^2(\Omega)}^2 : \mathbf{v} \in H_0(\operatorname{div}, \Omega), |\mathbf{v}(x)|_{\ell^{r^*}} \leq \lambda(x) \text{ f.a.a. } x \in \Omega \right\},$$

where  $\frac{1}{r} + \frac{1}{r^*} = 1$ .

## 5. Overlapping Domain Decomposition for Total Variation Denoising

### 5.2.3. Alternating Algorithm

Based on the above considerations we are now able to present the proposed alternating domain decomposition algorithm, stated in Algorithm 2.

---

**Algorithm 2** Alternating Version

---

```

Initialize:  $u_2^0 (= 0) \in L^2(\Omega)$ ,  $f_2^0 = 0 \in L^2(\Omega)$ 
for  $n = 0, 1, 2, \dots$  do
     $f_1^{n+1} = u_2^n - f_2^n + g$ 
     $u_1^{n+1} = \arg \min_{u_1 \in L^2(\Omega)} \frac{1}{2} \|u_1 - f_1^{n+1}\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} \theta_1 |Du_1|$ 
     $f_2^{n+1} = u_1^{n+1} - f_1^{n+1} + g$ 
     $u_2^{n+1} = \arg \min_{u_2 \in L^2(\Omega)} \frac{1}{2} \|u_2 - f_2^{n+1}\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} \theta_2 |Du_2|$ 
     $u^{n+1} = g - f_1^{n+1} - f_2^{n+1} + u_1^{n+1} + u_2^{n+1} (= u_2^{n+1})$ 
end for

```

---

**Lemma 5.7.** *The sequence  $(u^n)_n$  as well as  $(u_i^n)_n$ ,  $i = 1, 2$ , generated by Algorithm 2 are bounded in  $L^2(\Omega)$ . Moreover, we have*

$$\|u_2^n\|_{L^2(\Omega)} \geq \|u_1^{n+1}\|_{L^2(\Omega)} \geq \|u_2^{n+1}\|_{L^2(\Omega)} \geq \|u_1^{n+2}\|_{L^2(\Omega)} \quad \text{for all } n \geq 1. \quad (5.12)$$

*Proof.* By the relation to the predual problem for  $n \geq 1$  we have for  $i = 1, 2$ , that there exist  $\mathbf{p}_i^n \in K_i := \{\mathbf{p} \in H_0(\text{div}, \Omega) : |\mathbf{p}|_{\ell^2} \leq \alpha \theta_i \text{ a.e. on } \Omega\}$  such that

$$u_i^n = \text{div } \mathbf{p}_i^n + f_i^n \quad \text{and} \quad \langle (-\text{div})^* u_i^n, \mathbf{p}_i - \mathbf{p}_i^n \rangle_{H_0(\text{div}, \Omega)' \times H_0(\text{div}, \Omega)} \leq 0 \quad \forall \mathbf{p}_i \in K_i. \quad (5.13)$$

This means that for all  $n \geq 1$  and  $i = 1, 2$  such  $\mathbf{p}_i^n \in K_i$  solve

$$\arg \min_{\mathbf{p} \in K_i} \frac{1}{2} \|\text{div } \mathbf{p} + f_i^n\|_{L^2(\Omega)}^2.$$

Since  $f_1^{n+1} = u_2^n - f_2^n + g = \text{div } \mathbf{p}_2^n + g$  and  $f_2^{n+1} = u_1^{n+1} - f_1^{n+1} + g = \text{div } \mathbf{p}_1^{n+1} + g$  we have

$$\mathbf{p}_1^{n+1} \in \arg \min_{\mathbf{p}_1 \in K_1} \frac{1}{2} \|\text{div } \mathbf{p}_1 + \text{div } \mathbf{p}_2^n + g\|_{L^2(\Omega)}^2 \quad \text{and} \quad \mathbf{p}_2^{n+1} \in \arg \min_{\mathbf{p}_2 \in K_2} \frac{1}{2} \|\text{div } \mathbf{p}_2 + \text{div } \mathbf{p}_1^{n+1} + g\|_{L^2(\Omega)}^2$$

and hence

$$\frac{1}{2} \|\text{div } \mathbf{p}_1^n + \text{div } \mathbf{p}_2^n + g\|_{L^2(\Omega)}^2 \geq \frac{1}{2} \|\text{div } \mathbf{p}_1^{n+1} + \text{div } \mathbf{p}_2^n + g\|_{L^2(\Omega)}^2 \geq \frac{1}{2} \|\text{div } \mathbf{p}_1^{n+1} + \text{div } \mathbf{p}_2^{n+1} + g\|_{L^2(\Omega)}^2 \quad (5.14)$$

for all  $n \geq 1$ . Set  $\mathbf{p}^n := \mathbf{p}_1^n + \mathbf{p}_2^n$  and  $\tilde{\mathbf{p}}^{n+1} := \mathbf{p}_1^{n+1} + \mathbf{p}_2^n$  for all  $n \geq 1$ , then (5.14) yields

$$\frac{1}{2} \|\text{div } \mathbf{p}_1^n + g\|_{L^2(\Omega)}^2 \geq \frac{1}{2} \|\text{div } \tilde{\mathbf{p}}^{n+1} + g\|_{L^2(\Omega)}^2 \geq \frac{1}{2} \|\text{div } \mathbf{p}_1^{n+1} + g\|_{L^2(\Omega)}^2 \quad \forall n \geq 1.$$

Note, that  $\mathbf{p}^1 \in H_0(\text{div}, \Omega)$  and hence  $\|\text{div } \mathbf{p}^1 + g\|_{L^2(\Omega)}^2 < \infty$ . Then, by the monotonicity we obtain

$$\|\text{div } \tilde{\mathbf{p}}^{n+1} + g\|_{L^2(\Omega)} < \infty \quad \text{and} \quad \|\text{div } \mathbf{p}^n + g\|_{L^2(\Omega)} < \infty$$

## I. Subspace Correction Methods for Total Variation Minimization

for all  $n \geq 1$ . Moreover,

$$\|\mathbf{p}_i^n\|_{\mathbb{L}^2(\Omega)}^2 = \int_{\Omega} |p_i^{1,n}(x)|^2 dx + \int_{\Omega} |p_i^{2,n}(x)|^2 dx \leq \alpha^2 \int_{\Omega} |\theta_i(x)|^2 dx < \infty,$$

where  $\mathbf{p}_i^n = (p_i^{1,n}, p_i^{2,n})$ , and hence

$$\|\mathbf{p}^n\|_{\mathbb{L}^2(\Omega)} \leq \|\mathbf{p}_1^n\|_{\mathbb{L}^2(\Omega)} + \|\mathbf{p}_2^n\|_{\mathbb{L}^2(\Omega)} < \infty \quad \text{and} \quad \|\tilde{\mathbf{p}}^{n+1}\|_{\mathbb{L}^2(\Omega)} < \infty$$

for all  $n \geq 1$ , which shows the boundedness of  $(\mathbf{p}^n)_n$  and  $(\tilde{\mathbf{p}}^{n+1})_n$  in  $H_0(\operatorname{div}, \Omega)$ . Set  $f^{n+1} := f_1^{n+1} + f_2^{n+1} = \operatorname{div} \tilde{\mathbf{p}}^{n+1} + 2g$  and  $\tilde{u}^n := u_1^n + u_2^n = \operatorname{div} \mathbf{p}^n + f^n$  for all  $n \geq 1$ , then we have, that  $(f^n)_n$  and  $(\tilde{u}^n)_n$  are bounded in  $L^2(\Omega)$ . Consequently, the sequences  $(u^n)_n$  and  $(u_2^n)_n$ , see Algorithm 2, are bounded in  $L^2(\Omega)$ . Moreover, we get the boundedness of  $(u_1^n)_n$ , which is easily seen from the definition of  $\tilde{u}^n$ .

Due to the relation (5.13) we obtain from the monotonicity (5.14) that

$$\begin{aligned} \|u_1^n - f_1^n + u_2^n - f_2^n + g\|_{L^2(\Omega)}^2 &\geq \|u_1^{n+1} - f_1^{n+1} + u_2^n - f_2^n + g\|_{L^2(\Omega)}^2 \\ &\geq \|u_1^{n+1} - f_1^{n+1} + u_2^{n+1} - f_2^{n+1} + g\|_{L^2(\Omega)}^2 \end{aligned}$$

for all  $n \geq 1$ . Since we have  $u_2^n - f_2^n - f_1^{n+1} + g = 0$  and  $u_1^n - f_2^n - f_1^n + g = 0$  for all  $n \geq 1$ , see Algorithm 2, the latter inequalities yield (5.12).  $\square$

Since  $u^n = u_2^n$  for all  $n \geq 1$ , the previous lemma shows, that  $(u^n)_n$  possesses weakly converging subsequences. Moreover, from the monotonicity in Lemma 5.7 we even get, that the whole sequence  $(\|u^n\|_{L^2(\Omega)})_n$  converges monotonically to some limit. Note, that  $\|u^n\|_{L^2(\Omega)} = \|u^{n+1}\|_{L^2(\Omega)}$  does not imply  $u^n = u^{n+1}$ . Now, we are wondering if the weak limit of such a subsequence indeed solves the global minimization problem (5.1). In order to argue that, we need the boundedness of  $(f_i^n)_n$  in  $L^2(\Omega)$ . By Algorithm 2 we have

$$f_1^n = \sum_{k=1}^{n-1} (u_2^k - u_1^k) + u_2^0 - f_2^0 + g, \quad f_2^n = u_1^n + \sum_{k=1}^{n-1} (u_1^k - u_2^k) - u_2^0 + f_2^0 = \sum_{k=1}^{n-1} (u_1^{k+1} - u_2^k) + u_1^1$$

for all  $n \geq 1$ . Hence, if there exists a constant  $C > 0$ , which is independent of  $n$  such that  $\sum_{k=1}^{n-1} \|u_2^k - u_1^k\|_{L^2(\Omega)} \leq C < \infty$  for any  $n \geq 1$  (or in other words, if  $\|u_2^k - u_1^k\|_{L^2(\Omega)}$  converges to zero fast enough for  $k \rightarrow \infty$ , for instance,  $\|u_2^k - u_1^k\|_{L^2(\Omega)} \leq \frac{1}{k^\gamma}$  with  $\gamma > 1$ ), then  $(f_i^n)_n$  is bounded in  $L^2(\Omega)$ . Unfortunately, in the sequel we are only able to show that if  $(f_i^n)_n$  is bounded, then  $\lim_{n \rightarrow \infty} \|u_2^n - u_1^n\|_{L^2(\Omega)} = 0$ .

**Theorem 5.8.** *If there is an  $i \in \{1, 2\}$  such that  $(f_i^n)_n$  is bounded in  $L^2(\Omega)$ , then Algorithm 2 generates a sequence  $(u^n)_n$  which converges strongly in  $L^2(\Omega)$  to the unique minimizer of (5.1).*

*Proof.* We use the notation of the previous proof.

Since  $(f_i^n)_n$  is bounded in  $L^2(\Omega)$  for one  $i \in \{1, 2\}$ , we get by the proof of Lemma 5.7 and Algorithm 2 that also  $(f_{i^c}^n)$ ,  $i^c \in \{1, 2\} \setminus \{i\}$  is bounded. Then (5.13) implies that  $(\operatorname{div} \mathbf{p}_1^n)_n$  and  $(\operatorname{div} \mathbf{p}_2^n)_n$  are bounded in  $L^2(\Omega)$  and hence  $(\mathbf{p}_i^n)_n$  is bounded in  $H_0(\operatorname{div}, \Omega)$ .

By the boundedness of  $(\mathbf{p}_2^n)_n$  there is a subsequence  $(\mathbf{p}_2^{n_k})_k$  which converges weakly to a limit denoted by  $\mathbf{p}_2^\infty$ . Since  $(\mathbf{p}_2^{n_k-1})_k$  is bounded as well, there exists a subsequence  $(\mathbf{p}_2^{n_{k_j}-1})_j$  with

## 5. Overlapping Domain Decomposition for Total Variation Denoising

weak limit  $\tilde{\mathbf{p}}_2^\infty$ . Further  $(\mathbf{p}_1^{n_{k_j}})_j$  is bounded and has a subsequence  $(\mathbf{p}_1^{n_{k_jl}})$  which converges weakly to  $\mathbf{p}_1^\infty$ . Thus, from

$$\begin{aligned}\frac{1}{2} \|\operatorname{div} \mathbf{p}_1^{n-1} + \operatorname{div} \mathbf{p}_2^{n-1} + g\|_{L^2(\Omega)}^2 &\geq \frac{1}{2} \|\operatorname{div} \mathbf{p}_1^n + \operatorname{div} \mathbf{p}_2^{n-1} + g\|_{L^2(\Omega)}^2 \\ &\geq \frac{1}{2} \|\operatorname{div} \mathbf{p}_1^n + \operatorname{div} \mathbf{p}_2^n + g\|_{L^2(\Omega)}^2 \geq 0,\end{aligned}$$

for all  $n \geq 2$ , we get that

$$\frac{1}{2} \|\operatorname{div} \mathbf{p}_1^\infty + \operatorname{div} \mathbf{p}_2^\infty + g\|_{L^2(\Omega)}^2 = \frac{1}{2} \|\operatorname{div} \mathbf{p}_1^\infty + \operatorname{div} \tilde{\mathbf{p}}_2^\infty + g\|_{L^2(\Omega)}^2.$$

By the optimality  $\mathbf{p}_2^\infty \in \arg \min_{\mathbf{p}_2 \in K_2} \frac{1}{2} \|\operatorname{div} \mathbf{p}_1^\infty + \operatorname{div} \mathbf{p}_2 + g\|_{L^2(\Omega)}^2$  and the strict convexity of  $\|\cdot\|_{L^2(\Omega)}^2$  we obtain

$$\operatorname{div} \tilde{\mathbf{p}}^\infty = \operatorname{div} \mathbf{p}^\infty, \quad (5.15)$$

where  $\mathbf{p}^\infty = \mathbf{p}_1^\infty + \mathbf{p}_2^\infty$  and  $\tilde{\mathbf{p}}^\infty = \mathbf{p}_1^\infty + \tilde{\mathbf{p}}_2^\infty$ .

We show now that  $(u^n)_n$  has a weak accumulation point minimizing (5.1). A function  $u^* \in L^2(\Omega)$  solves (5.1) if and only if there exists  $\mathbf{p}^* \in H_0(\operatorname{div}, \Omega)$  with  $|\mathbf{p}^*(x)|_{\ell^2} \leq \alpha$  f.a.a.  $x \in \Omega$  such that

- (i)  $\operatorname{div} \mathbf{p}^* + g = u^*$  and
- (ii)  $\langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}^* + g), \mathbf{p} - \mathbf{p}^* \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} \leq 0$  for all  $\mathbf{p} \in H_0(\operatorname{div}, \Omega)$  with  $|\mathbf{p}(x)|_{\ell^2} \leq \alpha$  f.a.a.  $x \in \Omega$ .

By the equality in (5.13) we have that

$$u_1^n + u_2^n = \operatorname{div} \mathbf{p}_1^n + \operatorname{div} \mathbf{p}_2^n + f_1^n + f_2^n,$$

and hence

$$u^n = u_1^n + u_2^n - f_1^n - f_2^n + g = g + \operatorname{div} \mathbf{p}^n$$

for all  $n \geq 1$ . Since all occurring sequences are bounded we are able to take suitable subsequences with weak limits  $u^\infty$  and  $\mathbf{p}^\infty$  and obtain

$$u^\infty = g + \operatorname{div} \mathbf{p}^\infty. \quad (5.16)$$

By the inequality in (5.13) for  $n \geq 2$  we have that

$$\begin{aligned}\langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_1^n + f_1^n), \mathbf{p}_1 - \mathbf{p}_1^n \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_1 \in K_1 \\ \langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_2^n + f_2^n), \mathbf{p}_2 - \mathbf{p}_2^n \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_2 \in K_2,\end{aligned}$$

which is equivalent to

$$\begin{aligned}\langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_1^n + \operatorname{div} \mathbf{p}_2^{n-1} + g), \mathbf{p}_1 - \mathbf{p}_1^n \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_1 \in K_1 \\ \langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_2^n + \operatorname{div} \mathbf{p}_1^n + g), \mathbf{p}_2 - \mathbf{p}_2^n \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_2 \in K_2.\end{aligned}$$

Since all quantities are bounded, we take suitable subsequences  $n_k$ , such that  $\mathbf{p}_i^{n_k} \rightharpoonup \mathbf{p}_i^\infty$  for  $i = \{1, 2\}$  and  $\tilde{\mathbf{p}}^{n_k} \rightharpoonup \tilde{\mathbf{p}}^\infty$ , and we get

$$\begin{aligned}\langle (-\operatorname{div})^*(\operatorname{div} \tilde{\mathbf{p}}^\infty + g), \mathbf{p}_1 - \mathbf{p}_1^\infty \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_1 \in K_1 \\ \langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}^\infty + g), \mathbf{p}_2 - \mathbf{p}_2^\infty \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_2 \in K_2.\end{aligned}$$

## I. Subspace Correction Methods for Total Variation Minimization

By (5.15) and summing up the latter two inequalities yields

$$\langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}^\infty + g), \mathbf{p} - \mathbf{p}^\infty \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} \leq 0 \quad \forall \mathbf{p} := \mathbf{p}_1 + \mathbf{p}_2 \in K_1 + K_2.$$

Note, that  $|\mathbf{p}|_{\ell^2} = |\mathbf{p}_1 + \mathbf{p}_2|_{\ell^2} \leq |\mathbf{p}_1|_{\ell^2} + |\mathbf{p}_2|_{\ell^2} \leq \alpha$  a.e. on  $\Omega$ . On the contrary for any  $\mathbf{p} \in H_0(\operatorname{div}, \Omega)$  with  $|\mathbf{p}|_{\ell^2} \leq \alpha$  we set  $\mathbf{p}_i = \theta_i \mathbf{p}$  for  $i = \{1, 2\}$ , which implies that  $|\mathbf{p}_i|_{\ell^2} \leq \alpha \theta_i$  and  $\mathbf{p}_i \in K_i$ , thanks to (5.9)–(5.11). Hence we obtain

$$\langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}^\infty + g), \mathbf{p} - \mathbf{p}^\infty \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} \leq 0$$

for all  $\mathbf{p} \in H_0(\operatorname{div}, \Omega)$  with  $|\mathbf{p}|_{\ell^2} \leq \alpha$  a.e. on  $\Omega$ . This together with (5.16) and the fact that  $|\mathbf{p}^\infty|_{\ell^2} \leq \alpha$  a.e. on  $\Omega$  shows that  $u^\infty$  is a minimizer of (5.1). Since any converging subsequence of  $(u^n)_n$  converges weakly in  $L^2(\Omega)$  to the unique minimizer of (5.1), there is only one accumulation point and hence  $u^n \rightharpoonup u^\infty$ .

From the monotonicity (5.12) we get that  $\|u^n\|_{L^2(\Omega)} \rightarrow \|u^\infty\|_{L^2(\Omega)}$  monotonically. This together with the weak convergence  $u^n \rightharpoonup u^\infty$  yields the assertion, since  $\|u^n\|_{L^2(\Omega)}^2 = \|u^n - u^\infty\|_{L^2(\Omega)}^2 + 2(u^n - u^\infty, u^\infty) + \|u^\infty\|_{L^2(\Omega)}^2$ .  $\square$

**Remark 5.9.** *The above proof relies on the boundedness of  $(\mathbf{p}_i^n)_n$  in  $H_0(\operatorname{div}, \Omega)$ , which is attained if  $(f_i^n)_n$  is bounded in  $L^2(\Omega)$  for  $i = \{1, 2\}$ . However, in a finite dimensional setting, which is for example the situation when the considered problem is discretized, the boundedness of  $(\mathbf{p}_i^n)_n$  implies the boundedness of  $(\operatorname{div} \mathbf{p}_i^n)_n$  for  $i = 1, 2$ , which in turn yields the boundedness of  $(f_i^n)_n$ . Hence, in this situation the boundedness assumption on  $(f_i^n)_n$  always holds and thus can be dropped in Theorem 5.8. Consequently, in a discrete setting by the above considerations we can get a vague idea of the convergence order. In particular, there exists a  $\gamma > 1$  such that  $\|u_1^n - u_2^n\|_X \leq \frac{1}{n^\gamma}$  for all  $n \geq 1$ , where  $\|\cdot\|_X$  defines an appropriate discrete norm.*

We emphasize, Theorem 5.8 shows, that the whole sequence  $(u^n)_n$  generated by Algorithm 2 converges strongly to a minimizer of the global problem (5.1), thanks to the monotonicity property in Lemma 5.7. This monotonicity property also guarantees, that  $(u_1^n)_n$  converges strongly to the minimizer of (5.1), which we show next.

**Corollary 5.10.** *Suppose the assumption of Theorem 5.8 holds. Then  $(u_1^n)_n$  generated by Algorithm 2 converges strongly in  $L^2(\Omega)$  to the minimizer of (5.1).*

*Proof.* Since  $(u_1^n)_n$  is bounded in  $L^2(\Omega)$  and  $u_1^n = \operatorname{div} \tilde{\mathbf{p}}^n + g$  for all  $n \geq 2$ , there exists a subsequence such that  $u_1^{n_k} \rightharpoonup u_1^\infty := \operatorname{div} \tilde{\mathbf{p}}^\infty + g = \operatorname{div} \mathbf{p}^\infty + g = u^\infty$  (in  $L^2(\Omega)$ ), where we used (5.15). Since this is true for any convergent subsequence, we get  $u_1^n \rightharpoonup u^\infty$  for  $n \rightarrow \infty$ . The strong convergence follows from the monotonicity (5.12) together with the weak convergence.  $\square$

**Corollary 5.11.** *Suppose the assumption of Theorem 5.8 holds. Then  $(f_i^n)_n$ ,  $i = 1, 2$ , generated by Algorithm 2 converges strongly in  $L^2(\Omega)$ .*

*Proof.* By Algorithm 2 we have for all  $n \geq 0$ ,

$$f_1^{n+1} = u_2^n - f_2^n + g = u_2^n - u_1^n + f_1^n, \quad f_2^{n+1} = u_1^{n+1} - f_1^{n+1} + g = u_1^{n+1} - u_2^n + f_2^n$$

and hence

$$f_1^{n+1} - f_1^n = u_2^n - u_1^n, \quad f_2^{n+1} - f_2^n = u_1^{n+1} - u_2^n.$$

This together with Corollary 5.10 and the boundedness of  $(f_i^n)_n$ ,  $i = 1, 2$ , implies the assertion.  $\square$

## 5. Overlapping Domain Decomposition for Total Variation Denoising

### 5.2.4. Parallel Algorithm

The parallel version of the domain decomposition algorithm in Algorithm 2 is presented in Algorithm 3. We state a similar convergence result as for the sequential algorithm.

---

**Algorithm 3** Parallel Version

---

```

Initialize:  $v_i^0 = 0$  for  $i = 1, 2$ 
for  $n = 0, 1, 2, \dots$  do
     $f_1^{n+1} = v_2^n + g$ 
     $f_2^{n+1} = v_1^n + g$ 
     $u_1^{n+1} = \arg \min_{u_1 \in L^2(\Omega)} \frac{1}{2} \|u_1 - f_1^{n+1}\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} \theta_1 |Du_1|$ 
     $u_2^{n+1} = \arg \min_{u_2 \in L^2(\Omega)} \frac{1}{2} \|u_2 - f_2^{n+1}\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} \theta_2 |Du_2|$ 
     $v_1^{n+1} = \frac{v_1^n + u_1^{n+1} - f_1^{n+1}}{2}$ 
     $v_2^{n+1} = \frac{v_2^n + u_2^{n+1} - f_2^{n+1}}{2}$ 
     $u^{n+1} = g + v_1^{n+1} + v_2^{n+1} (= \frac{u_1^{n+1} + u_2^{n+1}}{2})$ 
end for

```

---

**Theorem 5.12.** Assume that  $(f_i^n)_n$  is bounded in  $L^2(\Omega)$  for  $i = 1, 2$ . Then Algorithm 3 generates a sequence  $(u^n)_n$ , which converges strongly in  $L^2(\Omega)$  to the minimizer of (5.1).

*Proof.* For the sake of clarity we present the proof in four steps.

**Step 1:** One shows by induction, as in [45, Proof of Lemma 3.4], that for  $i = 1, 2$  there exist  $\tilde{\mathbf{v}}_i^n \in K_i$  such that  $\operatorname{div} \tilde{\mathbf{v}}_i^n = v_i^n$  for all  $n \geq 0$ .

**Step 2:** Now let us show that  $(\mathbf{p}_i^{n+1})_n$  is bounded in  $H_0(\operatorname{div}, \Omega)$ . From the boundedness of  $(f_i^n)_n$  we directly get that  $(v_i^n)_n$  is bounded. We note that for all  $n \geq 0$

$$\mathbf{p}_i^{n+1} \in \arg \min_{p_i \in K_i} \|\operatorname{div} \mathbf{p}_i + f_i^{n+1}\|_{L^2(\Omega)}^2,$$

where  $f_i^{n+1}$  is given by Algorithm 3. Using this and the triangle inequality, we obtain

$$\begin{aligned}
\|v_1^{n+1} + v_2^{n+1} + g\|_{L^2(\Omega)} &= \left\| \frac{v_1^n + \operatorname{div} \mathbf{p}_1^{n+1}}{2} + \frac{v_2^n + \operatorname{div} \mathbf{p}_2^{n+1}}{2} + g \right\|_{L^2(\Omega)} \\
&\leq \frac{1}{2} (\|\operatorname{div} \mathbf{p}_1^{n+1} + v_2^n + g\|_{L^2(\Omega)} + \|v_1^n + \operatorname{div} \mathbf{p}_2^{n+1} + g\|_{L^2(\Omega)}) \quad (5.17) \\
&\leq \frac{1}{2} (\|\operatorname{div} \tilde{\mathbf{v}}_1^n + v_2^n + g\|_{L^2(\Omega)} + \|v_1^n + \operatorname{div} \tilde{\mathbf{v}}_2^n + g\|_{L^2(\Omega)}) \\
&= \|v_1^n + v_2^n + g\|_{L^2(\Omega)}
\end{aligned}$$

for all  $n \geq 0$ . Similarly one gets

$$\begin{aligned}
\|v_1^{n+1} + v_2^n + g\|_{L^2(\Omega)} &= \left\| \frac{v_1^n + \operatorname{div} \mathbf{p}_1^{n+1}}{2} + v_2^n + g \right\|_{L^2(\Omega)} \\
&\leq \frac{1}{2} (\|\operatorname{div} \mathbf{p}_1^{n+1} + v_2^n + g\|_{L^2(\Omega)} + \|v_1^n + v_2^n + g\|_{L^2(\Omega)}) \quad (5.18) \\
&= \|v_1^n + v_2^n + g\|_{L^2(\Omega)}
\end{aligned}$$

## I. Subspace Correction Methods for Total Variation Minimization

for all  $n \geq 0$ . By the monotonicity in (5.17) we have that  $\|v_1^n + v_2^n + g\|_{L^2(\Omega)} \leq \|g\|_{L^2(\Omega)}$ . Thus from (5.18) we obtain the boundedness in  $L^2(\Omega)$  for  $(\operatorname{div} \mathbf{p}_1^{n+1})_n$ , since

$$\|g\|_{L^2(\Omega)} \geq \|v_1^n + v_2^n + g\|_{L^2(\Omega)} \geq \|\operatorname{div} \mathbf{p}_1^{n+1} + v_2^n + g\|_{L^2(\Omega)} \geq \|\operatorname{div} \mathbf{p}_1^{n+1}\|_{L^2(\Omega)} - \|v_2^n\|_{L^2(\Omega)} - \|g\|_{L^2(\Omega)},$$

for all  $n \geq 0$ . An analog argument yields the boundedness of  $(\operatorname{div} \mathbf{p}_2^{n+1})_n$  in  $L^2(\Omega)$ .

Since  $\mathbf{p}_i^{n+1} \in K_i$  for all  $n \geq 0$ , we get  $\|\mathbf{p}_i^{n+1}\|_{L^2(\Omega)}^2 \leq \alpha^2 \|\theta_i\|_{L^2(\Omega)}^2 < \infty$  and hence  $(\mathbf{p}_i^{n+1})_n$  is bounded in  $H_0(\operatorname{div}, \Omega)$ .

**Step 3:** We show that weak accumulation points of  $(v_i^n)_n$  and  $(\operatorname{div} \mathbf{p}_i^{n+1})_n$  coincide. Since both sequences are bounded, there exist suitable subsequences such that  $(v_i^{n_k})_k$  and  $(\operatorname{div} \mathbf{p}_i^{n_k+1})_k$  have weak limits  $v_i^\infty$  and  $\operatorname{div} \mathbf{p}_i^\infty$ , respectively. Since  $\mathbf{p}_1^{n_k+1} \in \arg \min_{\mathbf{p}_1 \in K_1} \|\operatorname{div} \mathbf{p}_1 + v_2^{n_k} + g\|_{L^2(\Omega)}^2$  and  $\mathbf{p}_2^{n_k+1} \in \arg \min_{\mathbf{p}_2 \in K_2} \|\operatorname{div} \mathbf{p}_2 + v_1^{n_k} + g\|_{L^2(\Omega)}^2$  we get from (5.17)

$$\begin{aligned} \|\operatorname{div} \mathbf{p}_1^{n_k+1} + v_2^{n_k} + g\|_{L^2(\Omega)}^2 &\leq \|v_1^{n_k} + v_2^{n_k} + g\|_{L^2(\Omega)}^2 \quad \text{and} \\ \|\operatorname{div} \mathbf{p}_2^{n_k+1} + v_1^{n_k} + g\|_{L^2(\Omega)}^2 &\leq \|v_1^{n_k} + v_2^{n_k} + g\|_{L^2(\Omega)}^2. \end{aligned}$$

Passing in (5.17) for the suitable subsequence  $(n_k)_k$  to the limit, we obtain that

$$\begin{aligned} \|v_1^\infty + v_2^\infty + g\|_{L^2(\Omega)}^2 - \|\operatorname{div} \mathbf{p}_1^\infty + v_2^\infty + g\|_{L^2(\Omega)}^2 &= 0 \quad \text{and} \\ \|v_1^\infty + v_2^\infty + g\|_{L^2(\Omega)}^2 - \|\operatorname{div} \mathbf{p}_2^\infty + v_1^\infty + g\|_{L^2(\Omega)}^2 &= 0. \end{aligned}$$

By the strict convexity of  $\|\cdot\|_{L^2(\Omega)}^2$  and the optimality of  $\mathbf{p}_i^\infty$  we get

$$v_i^\infty = \operatorname{div} \mathbf{p}_i^\infty$$

for  $i = 1, 2$ .

**Step 4:** We are left by showing that the sequence  $(u^n)_n$  generated by Algorithm 3 converges to a solution of (5.1). We recall, that a function  $u^* \in L^2(\Omega)$  solves (5.1) if and only if there exists  $\mathbf{p}^* \in H_0(\operatorname{div}, \Omega)$  with  $|\mathbf{p}^*(x)|_{\ell^2} \leq \alpha$  f.a.a.  $x \in \Omega$  such that

- (i)  $\operatorname{div} \mathbf{p}^* + g = u^*$  and
- (ii)  $\langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}^* + g), \mathbf{p} - \mathbf{p}^* \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} \leq 0$  for all  $\mathbf{p} \in H_0(\operatorname{div}, \Omega)$  with  $|\mathbf{p}(x)|_{\ell^2} \leq \alpha$  f.a.a.  $x \in \Omega$ .

Since  $(v_i^n)_n$  is bounded in  $L^2(\Omega)$  we also have that  $(u^n)_n$  is bounded in  $L^2(\Omega)$ . For all  $n \geq 1$  we have from step 1 that  $v_i^n = \operatorname{div} \tilde{\mathbf{v}}_i^n$ , where  $(\tilde{\mathbf{v}}_i^n)_n$  is bounded in  $L^2(\Omega)$ , because  $\tilde{\mathbf{v}}_i^n \in K_i$ . Setting  $\tilde{\mathbf{v}}^n := \tilde{\mathbf{v}}_1^n + \tilde{\mathbf{v}}_2^n$  we can write

$$u^{n+1} = g + \operatorname{div} \tilde{\mathbf{v}}_1^{n+1} + \operatorname{div} \tilde{\mathbf{v}}_2^{n+1} = g + \operatorname{div} \tilde{\mathbf{v}}^{n+1}$$

for all  $n \geq 1$ . Then let us take a suitable subsequence such that  $u^{n_k+1} \rightarrow u^\infty$  and  $\mathbf{p}^{n_k+1} \rightarrow \mathbf{p}^\infty$  and passing to the limit yields

$$u^\infty = g + \operatorname{div} \tilde{\mathbf{v}}^\infty. \tag{5.19}$$

By the inequality in (5.13) we have that

$$\begin{aligned} \langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_1^{n+1} + f_1^{n+1}), \mathbf{p}_1 - \mathbf{p}_1^{n+1} \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_1 \in K_1 \\ \langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_2^{n+1} + f_2^{n+1}), \mathbf{p}_2 - \mathbf{p}_2^{n+1} \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_2 \in K_2 \end{aligned}$$

## 5. Overlapping Domain Decomposition for Total Variation Denoising

for all  $n \geq 0$ , which is equivalent to

$$\begin{aligned} \langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_1^{n+1} + v_2^n + g), \mathbf{p}_1 - \mathbf{p}_1^{n+1} \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_1 \in K_1 \\ \langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_2^{n+1} + v_1^n + g), \mathbf{p}_2 - \mathbf{p}_2^{n+1} \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_2 \in K_2. \end{aligned}$$

Since all quantities are bounded, we take suitable subsequences  $n_k$  such that  $\mathbf{p}_i^{n_k+1} \rightharpoonup \mathbf{p}_i^\infty$  for  $i = \{1, 2\}$  and  $v_i^{n_k} \rightharpoonup v_i^\infty = \operatorname{div} \mathbf{p}_i^\infty$  (see step 3) and we get

$$\begin{aligned} \langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_1^\infty + g), \mathbf{p}_1 - \mathbf{p}_1^\infty \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_1 \in K_1 \\ \langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}_2^\infty + g), \mathbf{p}_2 - \mathbf{p}_2^\infty \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} &\leq 0 \quad \forall \mathbf{p}_2 \in K_2. \end{aligned}$$

Summing up the latter two inequalities yields

$$\langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}^\infty + g), \mathbf{p} - \mathbf{p}^\infty \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} \leq 0 \quad \forall \mathbf{p} := \mathbf{p}_1 + \mathbf{p}_2 \in K_1 + K_2,$$

which is, following the same arguments as in the proof of Theorem 5.8, equivalent to

$$\langle (-\operatorname{div})^*(\operatorname{div} \mathbf{p}^\infty + g), \mathbf{p} - \mathbf{p}^\infty \rangle_{H_0(\operatorname{div}, \Omega)' \times H_0(\operatorname{div}, \Omega)} \leq 0$$

for all  $\mathbf{p} \in H_0(\operatorname{div}, \Omega)$  with  $|\mathbf{p}|_{\ell^2} \leq \alpha$  a.e. on  $\Omega$ . This together with (5.19) and the fact that  $|\mathbf{p}^\infty|_{\ell^2} \leq \alpha$  a.e. on  $\Omega$  shows that  $u^\infty$  is a minimizer of (5.1). By the uniqueness of the minimizer, any convergent subsequence of  $(u^n)_n$  converges to  $u^\infty$  and hence  $u^n \rightharpoonup u^\infty$  for  $n \rightarrow \infty$ .

From (5.17) we directly obtain  $\|u^n\|_{L^2(\Omega)} \geq \|u^{n+1}\|_{L^2(\Omega)}$ . This together with the weak convergence  $u^n \rightharpoonup u^\infty$  implies the strong convergence.  $\square$

**Remark 5.13.** Alternatively to the boundedness assumption on  $(f_i^n)_n$ ,  $i = 1, 2$ , in Theorem 5.12 we may assume that  $(v_i^n)_n$ ,  $i = 1, 2$ , is bounded in  $L^2(\Omega)$  instead. This would obviously imply the boundedness of  $(f_i^n)_n$  in  $L^2(\Omega)$ . However, we note again, that the boundedness assumption on  $(f_i^n)_n$ ,  $i = 1, 2$ , in Theorem 5.12 indeed holds in a finite dimensional setting, cf. Remark 5.9.

As for the sequential method, in the infinite dimensional case the boundedness of  $(f_i^n)_n$  in  $L^2(\Omega)$  is only ensured, if  $\|u_2^n - u_1^n\|_{L^2(\Omega)}$  converges to 0 sufficiently fast (for  $n \rightarrow \infty$ ).

**Proposition 5.14.** Let  $(u_i^n)_n$ ,  $i = 1, 2$  be generated by Algorithm 3. If there exists a constant  $\gamma > 1$  such that  $\|u_2^n - u_1^n\|_{L^2(\Omega)} \leq \frac{1}{n^\gamma}$  for any  $n \geq 1$ , then  $(v_i^n)_n$  and  $(f_i^n)_n$ ,  $i = 1, 2$ , are bounded in  $L^2(\Omega)$ .

*Proof.* By induction one shows that

$$v_1^n = \frac{1}{4} \sum_{i=1}^{n-1} (u_1^i - u_2^i) + \frac{1}{2} u_1^n - \frac{1}{2} g \quad \text{and} \quad v_2^n = \frac{1}{4} \sum_{i=1}^{n-1} (u_2^i - u_1^i) + \frac{1}{2} u_2^n - \frac{1}{2} g$$

for  $n \geq 1$ . By the triangle inequality and the boundedness of  $(u_i^n)_n$  in  $L^2(\Omega)$ , the assertion follows.  $\square$

## I. Subspace Correction Methods for Total Variation Minimization

### 5.2.5. Multi-Subdomain

The domain decomposition methods presented in Algorithm 2 and Algorithm 3 can be naturally extended to a multi-domain splitting. Let  $M \in \mathbb{N}$  be the number of overlapping subdomains and  $(\theta_i)_{i=1}^M$  a partition of unity with the properties

- (a')  $\sum_{i=1}^M \theta_i \equiv 1$  and  $\theta_i \geq 0$  a.e. on  $\bar{\Omega}$  for  $i = 1, 2, \dots, M$ ,
- (b')  $\text{supp}(\theta_i) \subset \bar{\Omega}_i$  for  $i = 1, 2, \dots, M$ ,
- (c')  $\theta_i \in H^1(\Omega) \cap C(\bar{\Omega})$  and  $\|\nabla \theta_i\|_{L^\infty(\Omega)} < \infty$  for  $i = 1, 2, \dots, M$ .

The multi-subdomain versions of Algorithm 2 and Algorithm 3 are stated in Algorithm 4 and Algorithm 5, respectively.

---

#### Algorithm 4 Alternating Multi-subdomain Version

---

```

Initialize:  $u_i^0 (= 0) \in L^2(\Omega)$ ,  $f_i^0 = 0 \in L^2(\Omega)$ ,  $i = 1, \dots, M$ 
for  $n = 0, 1, 2, \dots$  do
    for  $i = 1, \dots, M$  do
         $f_i^{n+1} = \sum_{j>i} (u_j^n - f_j^n) + \sum_{j<i} (u_j^{n+1} - f_j^{n+1}) + g$ 
         $u_i^{n+1} = \arg \min_{u_i \in L^2(\Omega)} \frac{1}{2} \|u_i - f_1^{n+1}\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} \theta_i |Du_i|$ 
    end for
     $u^{n+1} = g + \sum_{i=1}^M u_i^{n+1} - f_i^{n+1} (= u_M^{n+1})$ 
end for

```

---



---

#### Algorithm 5 Parallel Multi-subdomain Version

---

```

Initialize:  $v_i^0 = 0$  for  $i = 1, 2, \dots, M$ 
for  $n = 0, 1, 2, \dots$  do
     $f_i^{n+1} = \sum_{j \neq i} v_j^n + g$ ,  $i = 1, \dots, M$ 
     $u_i^{n+1} = \arg \min_{u_i \in L^2(\Omega)} \frac{1}{2} \|u_i - f_i^{n+1}\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} \theta_i |Du_i|$ ,  $i = 1, \dots, M$ 
     $v_i^{n+1} = \frac{(M-1)v_i^n + u_i^{n+1} - f_i^{n+1}}{M}$ ,  $i = 1, \dots, M$ 
     $u^{n+1} = g + \sum_{i=1}^M v_i^{n+1} (= \frac{\sum_{i=1}^M v_i^{n+1}}{M})$ 
end for

```

---

Similar to Lemma 5.7 and Theorem 5.12 we are able to prove that the sequence  $(u^n)_n$ , generated by Algorithm 4 or Algorithm 5, is bounded in  $L^2(\Omega)$ . Further, if  $(f_i^n)_n$  for  $i = 1, \dots, M$  is bounded in  $L^2(\Omega)$ , then we can again show as in Theorem 5.8 and in Theorem 5.12 that  $(u^n)_n$  converges strongly in  $L^2(\Omega)$  to the minimizer of (5.1). We recall, that in a finite dimensional setting, the boundedness of  $(f_i^n)_n$  is easily shown as above. In an infinite dimensional setting we only obtain the following result:

**Proposition 5.15.** *Assume that  $M \in \mathbb{N}$  is a fixed number.*

- (a) *Let  $(u_i^n)_n$ ,  $i = 1, \dots, M$  be generated by Algorithm 4. If there exists a constant  $\gamma > 1$  such that  $\|u_{i-1}^{n+1} - u_i^n\|_{L^2(\Omega)} \leq \frac{1}{n^\gamma}$  for any  $n \geq 1$ , then  $(f_i^n)_n$  is bounded in  $L^2(\Omega)$  for  $i \in \{1, \dots, M\}$ , where we use the convention that  $u_0^{n+1} := u_M^n$ .*

## 5. Overlapping Domain Decomposition for Total Variation Denoising

(b) Let  $(u_i^n)_n$ ,  $i = 1, \dots, M$  be generated by Algorithm 5. If there exists a constant  $\gamma > 1$  such that  $\|\sum_{\substack{j=1 \\ j \neq i}}^M (u_i^n - u_j^n)\|_{L^2(\Omega)} \leq \frac{1}{n^\gamma}$  for any  $n \geq 1$  and  $i \in \{1, \dots, M\}$ , then  $(v_i^n)_n$  and  $(f_i^n)_n$  are bounded in  $L^2(\Omega)$ .

*Proof.* (a) By induction one shows that

$$f_i^n = \sum_{\ell=1}^{n-1} (u_{i-1}^{\ell+1} - u_i^\ell) + u_{i-1}^1, \quad \text{for } i = 1, \dots, M \quad \text{and } n \geq 1,$$

where  $u_0^1 := g$ . The assertion follows then by the triangle inequality.

(b) Again by induction one shows that

$$v_i^n = \frac{1}{M^2} \sum_{k=1}^{n-1} \sum_{j \neq i}^M (u_i^k - u_j^k) + \frac{1}{M} u_i^n - \frac{1}{M} g, \quad \text{for } i = 1, \dots, M \quad \text{and } n \geq 1.$$

Since  $(u_i^n)_n$  is bounded in  $L^2(\Omega)$ , we obtain the assertion.  $\square$

Moreover, we emphasize, that for the sequential version, see Algorithm 4, we obtain as in Lemma 5.7 the monotonicity  $\|u_1^n\|_{L^2(\Omega)} \geq \|u_2^n\|_{L^2(\Omega)} \geq \dots \geq \|u_M^n\|_{L^2(\Omega)} \geq \|u_1^{n+1}\|_{L^2(\Omega)}$  for  $n \geq 1$ . Consequently, for  $i \in \{1, \dots, M\}$  the sequence  $(u_i^n)_n$  converges strongly to a minimizer of the global problem.

**Remark 5.16.** In the same manner as presented above we may construct domain decomposition methods for (5.1) with  $\int_\Omega |Du|_r$  for any  $1 \leq r < +\infty$ . By using the equivalence of norms in finite dimensions, in particular, that there exist positive constants  $\bar{C}_{r^*}, \underline{C}_{r^*}$  such that  $\underline{C}_{r^*}|v|_{\ell^{r^*}} \leq |v|_{\ell^2} \leq \bar{C}_{r^*}|v|_{\ell^{r^*}}$ , where  $\frac{1}{r} + \frac{1}{r^*} = 1$ , the convergence proofs follow the same lines as the proofs above.

**Remark 5.17.** Actually the proposed domain decomposition algorithms may be used to solve the more general problem

$$\arg \min_{u \in L^2(\Omega)} \frac{1}{2} \|Tu - g\|_{L^2(\Omega)}^2 + \alpha \int_\Omega |Du|, \quad (5.20)$$

where  $T : L^2(\Omega) \rightarrow L^2(\Omega)$  is a bounded linear operator, not annihilating constant functions, in order to guarantee existence of a minimizer of (5.20) [1].

Then, by using an operator splitting technique [16], a solution  $u^*$  of (5.20) for any given  $u^0 \in L^2(\Omega)$  is obtained by iteratively solving

$$u^{k+1} = \arg \min_{u \in L^2(\Omega)} \|u - (u^k + \frac{1}{\gamma} T^*(g - Tu^k))\|_{L^2(\Omega)}^2 + \alpha \int_\Omega |Du|, \quad \text{for } k \geq 0, \quad (5.21)$$

where  $\gamma > \|T\|^2$ , see for example [17, 18]. Since the minimization problem in (5.21) is of the form (5.1), i.e., a denoising type of problem, the proposed domain decomposition methods may be applied to solve it.

## I. Subspace Correction Methods for Total Variation Minimization

### 5.3. Subspace Minimization

Let us consider, for example, the subspace minimization with respect to  $u_1$ , i.e.,

$$u_1^{n+1} = \arg \min_{u_1 \in L^2(\Omega)} \frac{1}{2} \|u_1 - f_1^{n+1}\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} \theta_1 |Du_1|. \quad (5.22)$$

We present two different approaches on how to compute the solution of (5.22) by solving a minimization on  $\Omega_1$  only. The first approach first discretizes (5.22) and then restricts the optimization process to the subdomain  $\Omega_1$ , while the second approach restricts the minimization problem in an infinite dimensional setting before discretization. These two approaches lead to nearly equal but still different discrete subspace problems.

#### 5.3.1. Discretize Before Restriction

We start by introducing a discretization of the subproblems constituted by the above introduced domain decomposition methods. Therefore, let  $\Omega^h$  be a discrete rectangular image domain containing  $N_1 \times N_2$  pixels,  $N_1, N_2 \in \mathbb{N}$ . We approximate functions  $u$  by discrete functions, denoted by  $u^h$ . The considered function spaces are  $X = \mathbb{R}^{N_1 \times N_2}$  and  $Y = X \times X$ . For  $u^h \in X$  and  $\mathbf{p}^h = (p^{h,1}, p^{h,2}) \in Y$  we use the norms  $\|u^h\|_X := (\sum_{x \in \Omega^h} |u^h(x)|^2)^{1/2}$  and  $\|\mathbf{p}^h\|_Y^2 := \|p^{h,1}\|_X^2 + \|p^{h,2}\|_X^2$ . On  $\Omega^h$  the discrete gradient  $\nabla_{\Omega}^h : X \rightarrow Y$  and the discrete divergence  $\text{div}_{\Omega}^h : Y \rightarrow X$  are defined in a standard way by forward and backward differences such that  $\text{div}_{\Omega}^h = -(\nabla_{\Omega}^h)^*$ ; see for example [34].

The discretized version of (5.22) using the above notation and definitions is written as

$$u_1^{h,n+1} = \arg \min_{u_1^h \in X} \frac{1}{2} \|u_1^h - f_1^{h,n+1}\|_X^2 + \alpha \sum_{x \in \Omega^h} \theta_1^h(x) |\nabla_{\Omega}^h u_1^h(x)|_{\ell^2}. \quad (5.23)$$

Let  $\Omega^h$  be decomposed into overlapping subdomains  $\Omega_i^h$ ,  $i = 1, \dots, M$  such that  $\Omega^h = \bigcup_{i=1}^M \Omega_i^h$  and for any  $i \in \{1, \dots, M\}$  there exists at least one  $j \in \{1, \dots, M\} \setminus \{i\}$  such that  $\Omega_i^h \cap \Omega_j^h \neq \emptyset$ . Due to this splitting, we define  $X_1 := \mathbb{R}^{|\Omega_1^h|}$  and  $Y_1 = X_1 \times X_1$ , and accordingly the norms  $\|u^h\|_{X_1} := (\sum_{x \in \Omega_1^h} |u^h(x)|^2)^{1/2}$ ,  $\|\mathbf{p}^h\|_{Y_1}^2 := \|p^{h,1}\|_{X_1}^2 + \|p^{h,2}\|_{X_1}^2$  for  $u^h \in X_1$  and  $\mathbf{p}^h \in Y_1$ . Let  $u^h \in X$ , then by  $u_1^h|_{\Omega_1^h}$  we define the restriction of  $u^h$  to  $\Omega_1^h$  and consequently  $u_1^h|_{\Omega_1^h} \in X_1$ . Since  $\theta_1^h(x) = 0$  for all  $x \in \Omega^h \setminus \Omega_1^h$  we can write the above minimization problem as

$$u_1^{h,n+1} = \begin{cases} f_1^{h,n+1} & \text{in } \Omega^h \setminus \Omega_1^h \\ \arg \min_{u_1^h \in X_1} \frac{1}{2} \|u_1^h - f_1^{h,n+1}\|_{X_1}^2 + \alpha \sum_{x \in \Omega_1^h} \theta_1^h(x) |\nabla_{\Omega}^h u_1^h(x)|_{\ell^2} & \text{in } \Omega_1^h, \end{cases} \quad (5.24)$$

where  $u_1^h \in X$  is such that  $u_1^h(x) = f_1^{h,n+1}(x)$  for  $x \in \Omega^h \setminus \Omega_1^h$ . Hence, in order to obtain  $u_1^{h,n+1}$  only a minimization problem in  $\Omega_1^h$  has to be solved.

Let us now discuss how we algorithmically solve the optimization problem in (5.24). Due to the presents of the function  $\theta_1$ , respectively  $\theta_1^h$ , no standard total variation minimization technique can be used. Instead one may need to adapt one of these standard approaches to this more general functional. We note, that the minimization of locally weighted total variation have been already considered in the literature and an algorithm to solve a minimization

## 5. Overlapping Domain Decomposition for Total Variation Denoising

problem of the type as in (5.24) is already presented in [39]. In particular, in [39] the primal-dual algorithm of [10] is adapted to a locally weighted total variation regularization. This method requires, that the locally distributed weights are strictly positive, i.e.,  $\theta_1^h(x) > 0$  for every  $x \in \Omega_1^h$  in (5.24). The proposed domain decomposition algorithms and their theory require that  $\text{supp } \theta_1 \subseteq \Omega_1^h$ . Hence, in our discrete setting one may easily set  $\theta_1^h > 0$  in  $\Omega_1^h$  and  $\theta_1^h = 0$  in  $\Omega^h \setminus \Omega_1^h$ , which allows to use the algorithm provided in [39]. Nevertheless, we derive an algorithm for locally weighted total variation, which does not require the strict positivity, and hence may be used to solve (5.22) and (5.23) as well. Actually we adapt the split Bregman algorithm [30], which we are explaining next.

We first replace  $(\nabla_\Omega^h u_1^h)|_{\Omega_1^h}$  with  $\mathbf{d}_1^h$  and then enforce the constraint  $(\nabla_\Omega^h u_1^h)|_{\Omega_1^h} = \mathbf{d}_1^h$  by applying the Bregman iteration yielding

$$\begin{aligned} (u_{1,\Omega_1^h}^{h,k+1}, \mathbf{d}_1^{h,k+1}) &\in \arg \min_{u_1^h|_{\Omega_1^h}, \mathbf{d}_1^h} \frac{1}{2} \|u_1^h - f_1^{h,n+1}\|_{X_1}^2 + \alpha \sum_{x \in \Omega_1^h} \theta_1^h |\mathbf{d}_1^h|_{\ell^2} + \frac{\mu}{2} \|\mathbf{d}_1^h - (\nabla_\Omega^h u_1^h)|_{\Omega_1^h} - \mathbf{b}_1^k\|_{Y_1}^2 \\ \mathbf{b}_1^{k+1} &= \mathbf{b}_1^k + (\nabla_\Omega^h u_1^{h,k+1})|_{\Omega_1^h} - \mathbf{d}_1^{h,k+1} \end{aligned} \quad (5.25)$$

where  $\mu > 0$  is a Lagrange multiplier and  $\mathbf{b}_1^k \in Y_1$ . The obtained minimization problem in (5.25) is then iteratively minimized, first with respect to  $u_1^h|_{\Omega_1^h}$  and then to  $\mathbf{d}_1^h$ . Note, that  $\nabla_\Omega^h$  is a quite local operator, i.e., it effects only neighbouring pixels. Hence, by carefully considering the restriction to  $\Omega_1^h$  (i.e., we use Dirichlet boundary conditions on the interface between  $\Omega_1^h$  and  $\Omega^h \setminus \Omega_1^h$ ),  $u_{1,\Omega_1^h}^{h,k+1} \in X_1$  is obtained, as in [30], by solving a linear system only of size  $|\Omega_1^h|$ , see Section 5.3.1.1. The optimal value  $\mathbf{d}_1^{h,k+1}$  can be computed by a shrinkage formula [56], i.e.,

$$\mathbf{d}_1^{h,k+1} = \max \left\{ |(\nabla_\Omega^h u_1^{h,k+1})|_{\Omega_1^h} + \mathbf{b}_1^k|_{\ell^2} - \frac{\alpha}{\mu} \theta_1^h, 0 \right\} \frac{(\nabla_\Omega^h u_1^{h,k+1})|_{\Omega_1^h} + \mathbf{b}_1^k}{|(\nabla_\Omega^h u_1^{h,k+1})|_{\Omega_1^h} + \mathbf{b}_1^k|_{\ell^2}},$$

where we follow the convention that  $0 \cdot \frac{0}{0} = 0$ .

Since this adapted split Bregman method does not require the strict positivity of  $\theta_1$  and  $\theta_1^h$ , it is also able to solve (5.22) and (5.23) by just adjusting the respective quantities accordingly (i.e., without restricting to  $\Omega_1^h$ ).

**Remark 5.18** (Overlapping to nonoverlapping). *In a discrete setting the continuity assumption on  $\theta_i^h$ , for  $i = 1, \dots, M$ , is obsolete. Hence we may let the overlapping-size go to 0 yielding a nonoverlapping decomposition. That is*

$$\theta_i^h(x) = \begin{cases} 1 & \text{if } x \in \Omega_i^h \\ 0 & \text{else} \end{cases}$$

for  $i = 1, \dots, M$ . Then the subspace minimization problems read as

$$\arg \min_{u_i \in X_i} \frac{1}{2} \|u_i^h - f_i^{h,n+1}\|_{X_i}^2 + \alpha \sum_{x \in \Omega_i^h} |\nabla_\Omega^h u_i^h(x)|_{\ell^2},$$

$i = 1, \dots, M$ . Thus in a discrete setting, using this discretization and restriction approach, in the limit case of a nonoverlapping decomposition Algorithm 2 and Algorithm 3 become the Block Gauss-Seidel and Relaxed Block Jacobi method of [45], respectively.

## I. Subspace Correction Methods for Total Variation Minimization

### 5.3.1.1. Solving (5.25) with Respect to $u_{1,\Omega_1^h}^{h,k+1}$

In order to get the solution  $u_{1,\Omega_1^h}^{h,k+1}$  in (5.25), we only need to solve a linear system. We describe here how this linear system looks when the domain is split vertically into overlapping stripes, as in Section 5.4.1 below. Note, that only the size of the resulting linear problem depends on the number of subdomains  $M$ , but not its structure.

Let the image domain  $\Omega^h = \{x_1^1 < x_2^1 < \dots < x_{N_1}^1\} \times \{x_1^2 < x_2^2 < \dots < x_{N_2}^2\}$  be a rectangular domain consisting of  $N_1 \times N_2$  discrete pixels. Then we decompose  $\Omega^h$  into overlapping subdomains  $\Omega_i^h$ ,  $i = 1, \dots, M$ , such that  $\Omega_1^h = \{x_1^1 < \dots < x_L^1\} \times \{x_1^2 < \dots < x_{N_2}^2\}$  and  $\Omega^h \setminus \Omega_1^h = \{x_{L+1}^1 < \dots < x_{N_1}^1\} \times \{x_1^2 < \dots < x_{N_2}^2\}$ . Further we set  $\tilde{\Omega}_1^h = \{x_{L+1}^1\} \times \{x_1^2 < \dots < x_{N_2}^2\}$ . We define a restriction of the global operator  $\operatorname{div}_{\Omega}^h$  to the domain  $\Omega_1^h$  by the local discrete divergent operator  $\tilde{\operatorname{div}}_{\Omega_1}^h$  as

$$(\tilde{\operatorname{div}}_{\Omega_1}^h \mathbf{p}^h)(x_i^1, x_j^2) = \begin{cases} p^{h,1}(x_i^1, x_j^2) & \text{if } i = 1 \\ p^{h,1}(x_i^1, x_j^2) - p^{h,1}(x_{i-1}^1, x_j^2) & \text{if } 1 < i \leq L \end{cases} + \begin{cases} p^{h,2}(x_i^1, x_j^2) & \text{if } j = 1 \\ p^{h,2}(x_i^1, x_j^2) - p^{h,2}(x_i^1, x_{j-1}^2) & \text{if } 1 < j < N_2 \\ -p^{h,2}(x_i^1, x_{j-1}^2) & \text{if } j = N_2 \end{cases}$$

for every  $\mathbf{p}^h = (p^{h,1}, p^{h,2}) \in Y_1$ . Accordingly, we denote the associated Laplace operator by  $\tilde{\Delta}_{\Omega_1}^h$  defined as

$$\tilde{\Delta}_{\Omega_1}^h u^h(x_i^1, x_j^2) = \begin{cases} u^h(x_i^1, x_j^2) - u^h(x_{i+1}^1, x_j^2) & \text{if } i = 1 \\ 2u^h(x_i^1, x_j^2) - u^h(x_{i+1}^1, x_j^2) - u^h(x_{i-1}^1, x_j^2) & \text{if } 1 < i \leq L. \end{cases} + \begin{cases} u^h(x_i^1, x_j^2) - u^h(x_i^1, x_{j+1}^2) & \text{if } j = 1 \\ 2u^h(x_i^1, x_j^2) - u^h(x_i^1, x_{j-1}^2) - u^h(x_i^1, x_{j+1}^2) & \text{if } 1 < j < N_2 \\ u^h(x_i^1, x_j^2) - u^h(x_i^1, x_{j-1}^2) & \text{if } j = N_2 \end{cases}$$

With the above notations the optimality of  $u_{1,\Omega_1^h}^{h,k+1}$  in (5.25) is equivalent to the solution  $u_1^h|_{\Omega_1^h}$  of the following boundary value problem

$$u_1^h(x) - \mu \tilde{\Delta}_{\Omega_1}^h u_1^h(x) = f_1^{h,n+1}(x) + \tilde{\operatorname{div}}_{\Omega_1}^h(\mathbf{b}_1^k - \mathbf{d}_1^h)(x), \quad x \in \Omega_1^h \quad (5.26)$$

$$u_1^h(x) = f_1^{h,n+1}(x), \quad x \in \tilde{\Omega}_1^h. \quad (5.27)$$

The system (5.26)-(5.27) may be also written as a linear system

$$A^h v^h = b^h,$$

where the vector  $v^h \in \mathbb{R}^{N_1 L}$  has the values  $u_1^h|_{\Omega_1^h}$  as its components arranged in a particular order,  $A^h \in \mathbb{R}^{N_1 L \times N_1 L}$  and  $b^h \in \mathbb{R}^{N_1 L}$  mimicking the associated matrix constituted from the left-hand side and the right-hand side of (5.26) together with the boundary conditions (5.27), respectively.

We remark, that similar considerations yield the linear system for obtaining  $u_{i,\Omega_i^h}^{h,k+1}$ ,  $i = 2, \dots, M$ .

## 5. Overlapping Domain Decomposition for Total Variation Denoising

### 5.3.2. Restrict Before Discretization

Let us turn back to the infinite dimensional subdomain problem (5.22). Since the partition of unity is such that  $\text{supp } \theta_1 \subseteq \Omega_1^h$ , we have  $\int_{\Omega} \theta_1 |Du_1| = \int_{\Omega_1} \theta_1 |Du_1|$ , cf. Lemma 5.2. Hence, by the optimality of  $u_1^{n+1}$  we get  $f_1^{n+1} - u_1^{n+1} \in \partial \alpha \int_{\Omega_1} \theta_1 |Du_1^{n+1}|$ . That is

$$(f_1^{n+1} - u_1^{n+1}, v - u_1^{n+1}) + \alpha \int_{\Omega_1} \theta_1 |Du_1^{n+1}| \leq \alpha \int_{\Omega_1} \theta_1 |Dv| \quad \forall v \in L^2(\Omega).$$

This inequality holds if

$$\begin{aligned} \int_{\Omega \setminus \Omega_1} (f_1^{n+1} - u_1^{n+1})(v - u_1^{n+1}) dx &\leq 0 \quad \text{and} \\ \int_{\Omega_1} (f_1^{n+1} - u_1^{n+1})(v - u_1^{n+1}) dx + \alpha \int_{\Omega_1} \theta_1 |Du_1^{n+1}| &\leq \alpha \int_{\Omega_1} \theta_1 |Dv| \end{aligned}$$

for all  $v \in L^2(\Omega)$ . Hence  $u_1^{n+1}$  fulfilling these two latter inequalities is a minimizer of the subspace minimization problem (5.22). By the uniqueness of the minimizer, see Proposition 5.3, we therefore obtain that

$$u_1^{n+1} = \begin{cases} f_1^{n+1} & \text{in } \Omega \setminus \Omega_1 \\ \arg \min_{u_1 \in L^2(\Omega_1)} \frac{1}{2} \|u_1 - f_1^{n+1}\|_{L^2(\Omega_1)}^2 + \alpha \int_{\Omega_1} \theta_1 |Du_1| & \text{in } \Omega_1. \end{cases} \quad (5.28)$$

Thus, generating the minimizer of (5.22) results in solving an optimization problem in  $\Omega_1$  only. As above, due to the presents of the function  $\theta_1$ , usual total variation minimization techniques cannot be used. For solving the optimization problem in (5.28), we want to use again the split Bregman method adapted to locally weighted total variation. Therefore, we discretize (5.28) by using the above notations and definitions yielding

$$u_1^{h,n+1} = \begin{cases} f_1^{h,n+1} & \text{in } \Omega_2^h \setminus \Omega_1^h \\ \arg \min_{u_1^h \in X_1} \frac{1}{2} \|u_1^h|_{\Omega_1^h} - f_1^{h,n+1}\|_{X_1}^2 + \alpha \sum_{x \in \Omega_1^h} \theta_1^h(x) |\nabla_{\Omega_1}^h(u_1^h|_{\Omega_1^h}(x))|_{\ell^2} & \text{in } \Omega_1^h, \end{cases} \quad (5.29)$$

where  $\nabla_{\Omega_1}^h$  denotes the standard gradient on  $\Omega_1^h$  with zero Neumann boundary conditions on  $\partial \Omega_1$ , cf. Section 5.3.1. Similar as above, one derives the adapted split Bregman algorithm, presented in Algorithm 6, where  $\Delta_{\Omega_1}^h$  denotes the respective standard discrete Laplace operator on  $\Omega_1^h$ .

Let us mention that all the results presented in Section 5.3.1 and Section 5.3.2 hold symmetrically for the minimization with respect to  $u_i$ ,  $i = 2, \dots, M$ , and that the notations should be just adjusted accordingly.

### 5.3.3. Comparison of the Two Restriction Approaches

If we compare (5.24) with (5.29), we observe, that the two above discussed restriction and discretization approaches lead to quite similar optimization problems, while the difference lies in the different discrete gradient operators. This difference indeed has a significant impact. While the “discretize before restriction” approach (DbR) allows for a nonoverlapping domain

## I. Subspace Correction Methods for Total Variation Minimization

---

**Algorithm 6** Split Bregman

---

```

Initialize:  $\mathbf{d}_1^{h,0} = 0 = \mathbf{b}_1^0$  and  $k = 0$ 
while stopping criterion does not hold do
    Solve  $(I - \mu\Delta_{\Omega_1}^h)u_1^{h,k+1}|_{\Omega_1^h} = f_1^{h,n+1} - \mu \operatorname{div}_{\Omega_1}^h(\mathbf{d}_1^{h,k} - \mathbf{b}_1^k)$ 
     $\mathbf{d}_1^{h,k+1} = \max\{|\nabla_{\Omega_1}^h(u_1^{h,k+1}|_{\Omega_1^h}) + \mathbf{b}_1^k|_{\ell^2} - \frac{\alpha}{\mu}\theta_1^h, 0\} \frac{\nabla_{\Omega_1}^h(u_1^{h,k+1}|_{\Omega_1^h}) + \mathbf{b}_1^k}{|\nabla_{\Omega_1}^h(u_1^{h,k+1}|_{\Omega_1^h}) + \mathbf{b}_1^k|_{\ell^2}}$ 
     $\mathbf{b}_1^{k+1} = \mathbf{b}_1^k + \nabla_{\Omega_1}^h(u_1^{h,k+1}|_{\Omega_1^h}) - \mathbf{d}_1^{h,k+1}$ 
     $k = k + 1$ 
end while

```

---

decomposition, see Remark 5.18, the “restrict before discretization” approach (RbD) does not support this situation. This is due to the fact, that the restriction process in the latter approach is done in the continuous setting, where the continuity of  $\theta_i$  is necessary. Moreover, from (5.29) we see that in case of a nonoverlapping decomposition, due to the “local” operator  $\nabla_{\Omega_i}^h$ , no information from outside of  $\Omega_i$  is entering  $\Omega_i$ , which would result in a wrong behaviour, see Figure 5.1. In particular in Figure 5.1 we contrast the reconstructions of the two approaches for a nonoverlapping decomposition. The figures show, as expected, that the “restrict before discretization” approach creates an artificial edge at the interface of the subdomains, while the “discretize before restriction” approach still works correct.

However, in case of an overlapping decomposition, for which the proposed algorithms are actually constructed, both approaches seem to generate a very alike reconstruction, see Figure 5.2 where the overlapping size is  $512 \times 20$  pixels. Moreover, performing several experiments, for both approaches the overall behaviour seems to be very similar, see Figure 5.3 for the example in Figure 5.2. More precisely, in all our experiments we observe that the sequential domain decomposition algorithm using the “restrict before discretization” approach always needs at most the same amount of iterations as the other approach, while in most of the experiments the algorithm even terminates one iteration earlier when using the “restrict before discretization” approach. In particular in Figure 5.3 we see, that with the “restrict before discretization” approach only 17 iterations till termination are needed while with the “discretize before restriction” approach 18 iterations are needed. In our numerical experiments in Section 5.4 we exclusively use the “first restrict and then discretize” approach, which seems theoretically more consistent, is finally even slightly easier to implement, since on the subdomains no specific Dirichlet boundary conditions have to be considered, and seems to be slightly faster (or at least not slower) than the other approach.

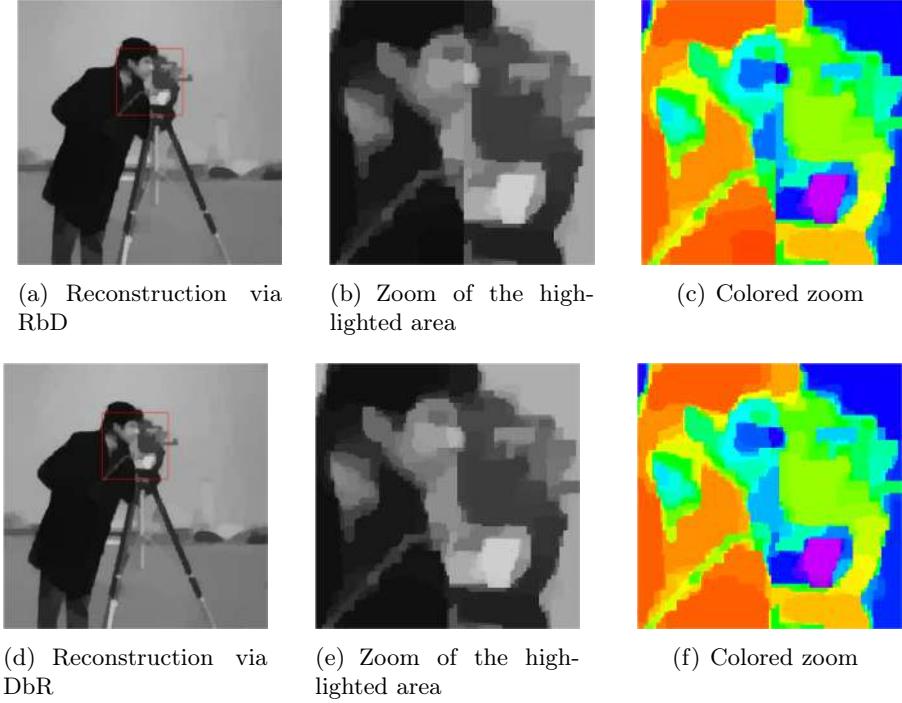
### 5.3.4. Implementation Issues

We recall that in each iteration of the proposed parallel domain decomposition algorithm we set

$$v_i^{n+1} = \frac{(M-1)v_i^n + u_i^{n+1} - f_i^{n+1}}{M}$$

for  $i = 1, \dots, M$ , see Algorithm 5. Note, that by (5.28) we have,  $u_i^{n+1} = f_i^{n+1}$  in  $\Omega \setminus \Omega_i$ . Since  $v_i^0 = 0$ , we iteratively obtain that  $\operatorname{supp} v_i^n \subset \Omega_i$  for all  $n \in \mathbb{N}$ . Thus, in our implementation we need to update  $v_i^n$  in  $\Omega_i$  only. Consequently only  $u_i^{n+1}$  in  $\Omega_i$  is needed, which is obtained

## 5. Overlapping Domain Decomposition for Total Variation Denoising



**Figure 5.1:** Reconstruction of an image of size  $512 \times 512$  pixels corrupted by additive Gaussian white noise with  $\sigma = 0.3$  using Algorithm 2 and a nonoverlapping splitting into 2 domains. In the first row we show the result via the “restrict before discretization” approach (RbD), while the second row shows the result via the “discretize before restriction” approach (DbR). In (b) and (e) we zoomed in on the in (a) and (d) highlighted area. In order to visualize the difference of the reconstructions, we color in (c) and (f) the zoomed area.

by solving an optimisation problem restricted so  $\Omega_i$ , see (5.28). Moreover, since  $u^{n+1} = \frac{\sum_{i=1}^M u_i^{n+1}}{M} = g + \sum_{i=1}^M v_i^{n+1}$ , we do not even need  $u_i^{n+1}$  for updating  $u^{n+1}$  and hence  $u_i^{n+1}$  does not have to be communicated, in case of a multi-processor implementation. Hence, in each subdomain we only need (to compute) quantities which are restricted to this subdomain, and the update  $u_i^{n+1} = f_i^{n+1}$  in  $\Omega \setminus \Omega_i$  is not performed at all.

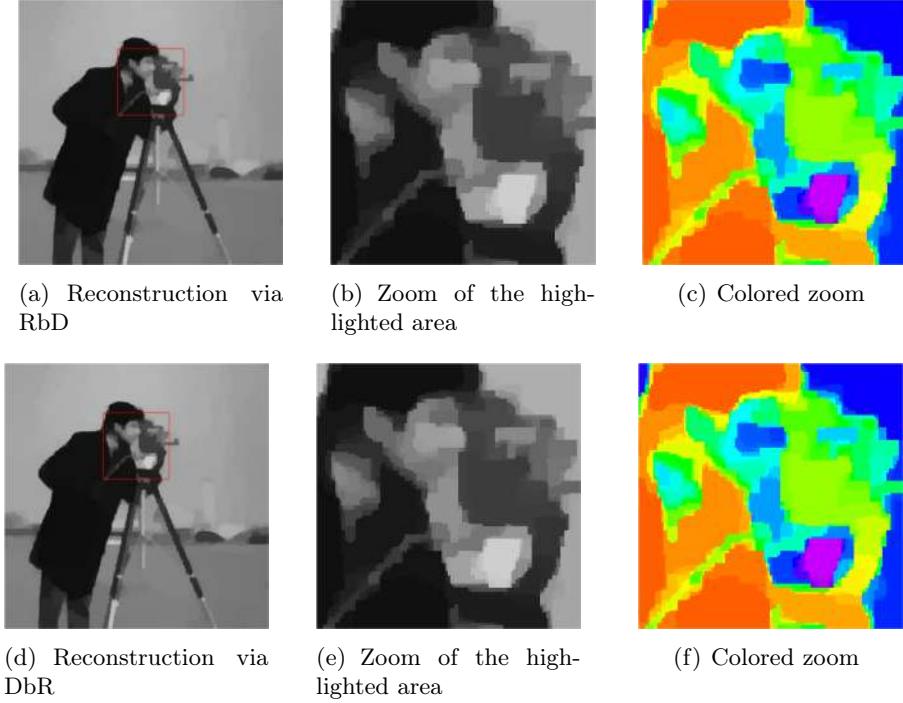
For the alternating domain decomposition algorithm we obtain the same. In particular, since  $u_i^n - f_i^n = 0$  in  $\Omega \setminus \Omega_i$ , we conclude that

$$u^n = g + \sum_{i=1}^M (u_i^n|_{\Omega_i} - f_i^n|_{\Omega_i}).$$

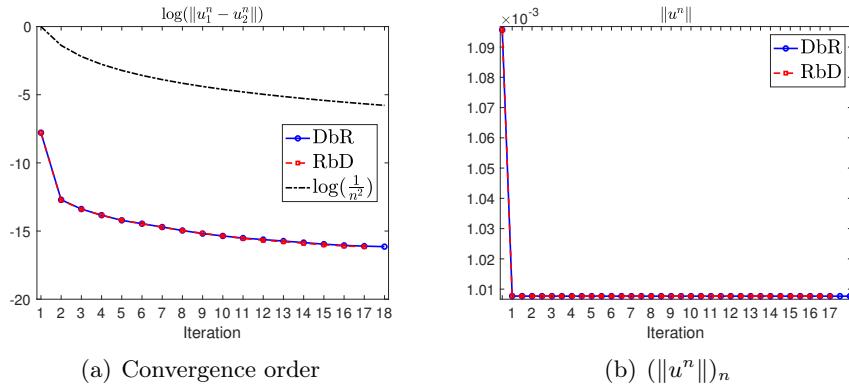
Note, that the same argumentation also holds in a discrete setting due to (5.24) and (5.29).

In order to speed up the minimization procedure in the subdomains, we only initialize in the first outer iteration the split Bregman algorithm as described in Algorithm 6. Later we use the approximation of the previous iterate. More precisely, let  $\tilde{u}_1^{h,n}$  be the solution of the split Bregman iteration in the  $n$ -th outer iteration and  $\tilde{\mathbf{d}}_1^{h,n}$ ,  $\tilde{\mathbf{b}}_1^n$  the associated variables. Then in the next iteration we initialize the split Bregman iteration by  $\mathbf{d}_1^{h,0} = \tilde{\mathbf{d}}_1^{h,n}$ , and  $\mathbf{b}_1^0 = \tilde{\mathbf{b}}_1^n$ .

## I. Subspace Correction Methods for Total Variation Minimization



**Figure 5.2:** Reconstruction of an image of size  $512 \times 512$  pixels corrupted by additive Gaussian white noise with  $\sigma = 0.3$  using Algorithm 2 and an overlapping splitting into 2 domains. In the first row we show the result via the “restrict before discretization” approach (RbD), while the second row shows the result via the “discretize before restriction” approach (DbR). In (b) and (e) we zoomed in on the in (a) and (d) highlighted area. In order to visualize possible differences of the reconstructions, we color in (c) and (f) the zoomed area.



**Figure 5.3:** Performance of Algorithm 2 using the “restrict before discretization” approach (RbD) and “discretize before restriction” approach (DbR).

## 5. Overlapping Domain Decomposition for Total Variation Denoising

### 5.4. Experiments

In the following section we present numerical experiments for the proposed sequential and parallel algorithms for image denoising. The value of the parameter  $\alpha$  is chosen arbitrarily and is not optimized in any way. For automatically choosing the regularization parameter  $\alpha$  in (5.1) we refer the reader to [39] and references therein. As subdomain solver we use Algorithm 6 with  $\mu = \frac{3}{\alpha}$ , which is terminated as soon as

$$\frac{\|u_i^{h,k} - u_i^{h,k-1}\|_{\ell^1}}{\|u_i^{h,k}\|_{\ell^1}} \leq \text{tol} \quad (5.30)$$

holds in subdomain  $i \in \{1, \dots, M\}$  for the first time, where tol is a predefined tolerance.

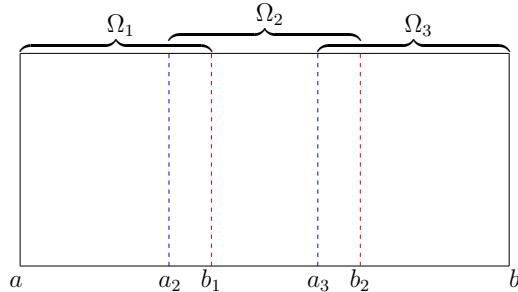
All the computations presented are done in Matlab on a MacBook Pro with 2.5 GHz Intel Core i7 processor (possesses 4 cores).

#### 5.4.1. Domain Decomposition

For simplicity let  $\Omega \subset \mathbb{R}^2$  be a rectangular domain  $[a, b] \times [c, d]$ , with  $a < b$  and  $c < d$ . Then we decompose  $\Omega$  into  $M \in \mathbb{N}$  subdomains  $\Omega_i$  such that

$$\Omega_i = [a_i, b_i] \times [c, d] \quad \text{for } i = 1, \dots, M$$

where  $a =: a_1 < a_2 < b_1 < a_3 < b_2 < \dots < b_{M-1} < b_M := b$ , see Figure 5.4 for  $M = 3$ . The overlapping width is then  $w_i = |b_i - a_{i+1}|$ .



**Figure 5.4:** Overlapping domain decomposition of  $\Omega$  into  $\Omega_1$ ,  $\Omega_2$  and  $\Omega_3$ .

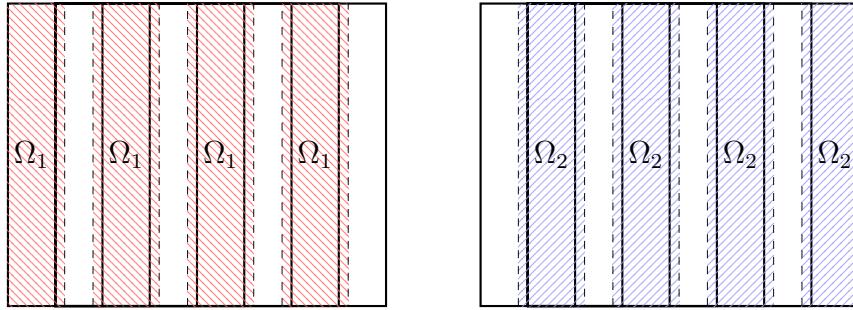
The auxiliary function  $\theta_i$ ,  $i = 1, \dots, M$ , is chosen as

$$\theta_i(x) = \begin{cases} 1 & \text{if } x \in \Omega_i \setminus (\Omega_{i-1} \cup \Omega_{i+1}) \\ \frac{b_i - x}{b_i - a_{i+1}} & \text{if } x \in \Omega_i \cap \Omega_{i+1} \\ \frac{x - a_i}{b_{i-1} - a_i} & \text{if } x \in \Omega_i \cap \Omega_{i-1} \\ 0 & \text{else (if } x \in \Omega \setminus \Omega_i) \end{cases} \quad \text{for } i = 1, \dots, M,$$

where  $\Omega_0 = \Omega_{M+1} = \emptyset$ . Compare with Figure 5.6 and Figure 5.7 for a decomposition into 2 and 3 domains, respectively.

In the parallel multi-subdomain method (Algorithm 5) for any  $n \geq 0$  the value  $v_i^{n+1}$  is obtained by a weighted sum of the previous iterate  $v_i^n$  and  $u_i^{n+1} - f_i^{n+1}$ , whereas the latter

## I. Subspace Correction Methods for Total Variation Minimization



**Figure 5.5:** Partitioning of an overlapping decomposition by colouring technique.

term is weighted by  $\frac{1}{M}$ . This weight tends to zero as the number of subdomains  $M$  grows, which may lead to a crucial decrease of the convergence speed of the algorithm. In order to overcome this behaviour, we may use a so-called *colouring technique*; see e.g. [53]. That is, the image domain  $\Omega$  is partitioned into a fixed number  $M_c$  of classes of overlapping subdomains, whereby each class is coloured by a different colour, i.e.,

$$\Omega = \bigcup_{j=1}^{M_c} \Omega_j,$$

where  $\Omega_j$  is the union of disjoint subdomains with the same colour. An example of an overlapping decomposition of a rectangular domain  $\Omega$  into 8 subdomains coloured by 2 different colours is depicted in Figure 5.5. We note, that in general the disjoint domains with same colour cannot be solved in parallel without introducing additional new constraints, as the following example, borrowed from Warga [57], shows.

**Example 5.19.** Let  $V := [0, 1] \times \{0\} \times [0, 1]$ ,  $V_1 := \{(c, 0, 0) \mid c \in [0, 1]\}$ ,  $V_3 := \{(0, 0, c) \mid c \in [0, 1]\}$ , and  $\varphi : V \rightarrow \mathbb{R}$  given by  $\varphi(x) = |x_1 - x_3| - \min\{x_1, x_3\}$ , where  $x = (x_1, x_2, x_3)$ . We have that  $\mathbf{0} = \arg \min_{x_i \in V_i} \varphi(x)$  for  $i \in \{1, 3\}$ , while  $(1, 0, 1) = \arg \min_{x \in V} \varphi(x)$ .

However, if the problem is additively separable with respect to the considered disjoint decomposition, then the problem can be solved easily in parallel on the disjoint domains. Since this property holds for our considered subdomain problems with respect to the disjoint domains with same colour, a partitioning with colouring technique changes the update of  $v_i^{n+1}$  to

$$v_i^{n+1} = \frac{(M_c - 1)v_i^n + u_i^{n+1} - f_i^{n+1}}{M_c}, \quad \text{for } n \geq 0,$$

where  $M_c \leq M$ . In particular if  $M$  is large, then  $M_c \ll M$ . For instance, decomposing and colouring as in Figure 5.5, keeps  $M_c = 2$  for any number of subdomains  $M > 1$ .

### 5.4.2. Sequential Algorithm

The scope of this section is to illustrate by simple examples the main properties of the algorithms, as proven in our theoretical analysis. In particular, we investigate the convergence order of  $\|u_{i-1}^{n+1} - u_i^n\|_{L^2(\Omega)}$  for  $i = 1, \dots, M$  and we demonstrate the monotonicity (5.12). Moreover, we emphasize the robustness in correct computing minimizers independently of the

## 5. Overlapping Domain Decomposition for Total Variation Denoising

size of overlapping regions and of the number of subdomains. As tolerance for the subspace solver we set  $\text{tol} = 10^{-6}$ .

In Figure 5.6 we show an image corrupted by additive Gaussian noise with standard deviation  $\sigma = 0.3$  and zero mean. This image is then reconstructed by means of total variation minimization. That is, the reconstruction is the solution of (5.1) with  $\alpha = 0.5$ . In order to obtain a minimizer of (5.1) we utilize the proposed alternating domain decomposition method, see Algorithm 2 and Algorithm 4. We test our algorithms for different numbers of subdomains, in particular we present results for  $M \in \{2, 3, 6\}$ , whereas the domain is split vertically, as shown in Figure 5.6 and Figure 5.7 for an overlapping decomposition into 2 and 3 domains respectively. There the red and blue lines indicate a right and left (inner) boundary of a subdomain respectively. The domain decomposition algorithm is terminated as soon as

$$\|u_M^n - u_{M-1}^n\|_{L^2(\Omega)} \leq 10^{-7}$$

for the first time, which indicates that nearly no changes are to be expected. This stopping criterion makes sense, since by Theorem 5.8 and Corollary 5.10, together with their extension to the multi-domain case, any converging subsequence of  $(u_i^n)_n$ ,  $i = 1, \dots, M$ , has the same limit, minimizing (5.1). The reconstructions of Figure 5.6(a) and Figure 5.7(a) obtained via the proposed overlapping domain decomposition algorithm, see Algorithm 4, where the overlapping regions are of size  $256 \times 80$  pixels, are shown in Figure 5.6(b) and Figure 5.7(b) respectively.

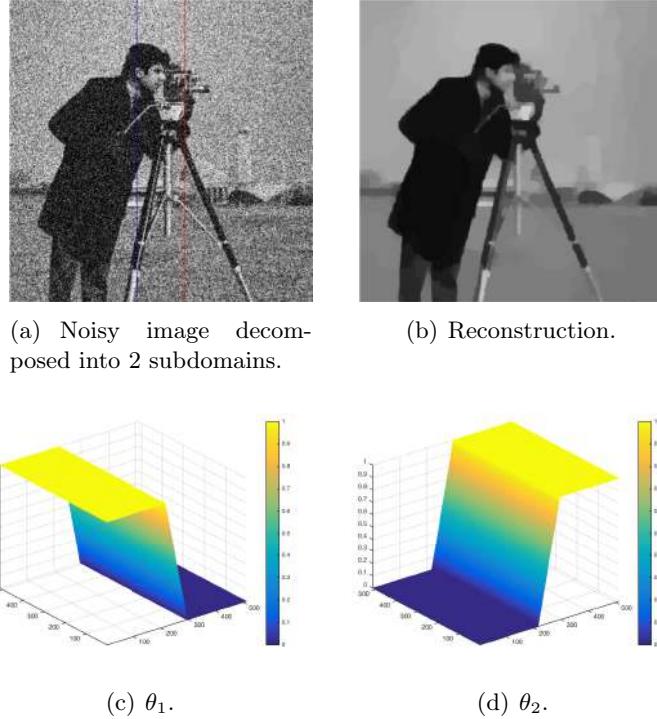
For a decomposition into  $M = 2$  subdomains, we depict in Figure 5.8(a) for different sizes of overlapping regions the decay of  $(\|u_2^n - u_1^n\|_{L^2(\Omega)})_n$ . In particular, we test for overlapping regions of size  $256 \times 20$  pixels,  $256 \times 80$  pixels, and  $256 \times 120$  pixels. We detect, that for  $\gamma = \frac{3}{2}$  we have  $\|u_2^n - u_1^n\|_{L^2(\Omega)} \leq \frac{1}{n^\gamma}$  for all  $n$ . This not only gives us an idea of the convergence order of the sequential domain decomposition algorithm, but also justifies the boundedness of  $(f_i^n)_n$ , which is actually anyway theoretically ensured in a discrete setting, cf. Remark 5.9. Moreover, we observe that the larger the overlapping region the more iterations are needed till termination. The monotonicity (5.12) is depicted in Figure 5.8(b), where we use the notation  $u^{n-1/2} := u_1^n$  and  $u^n := u_2^n$  for all  $n \in \mathbb{N}$ . We observe, that after one outer iteration  $\|u^n\|_{L^2(\Omega)}$  (where  $n = 1, 1 + \frac{1}{2}, 2, \dots$ ) does not change visibly anymore.

For  $M = 3$  and  $M = 6$ , we depict in Figure 5.9 the performance of the sequential domain decomposition algorithm for overlapping sizes of  $512 \times 40$  pixels. In Figure 5.9(a) and Figure 5.9(b) we present the decay of the norms  $\|u_{i-1}^{n+1} - u_i^n\|_{L^2(\Omega)}$ ,  $i = 1, \dots, M$ , with respect to the outer iteration  $n$ . As for the case  $M = 2$ , we observe, that  $\|u_{i-1}^{n+1} - u_i^n\|_{L^2(\Omega)} \leq \frac{1}{n^{3/2}}$  for  $i = 1, \dots, M$ , showing that the sequences  $(f_i^n)_n$  are indeed bounded. Using the notation  $u^{n+i/M} := u_i^{n+1}$  for  $n = 0, 1, 2, \dots$  and  $i = 1, \dots, M$ , we depict in Figure 5.9(c) and 5.9(d) the monotonicity of  $\|u^n\|_{L^2(\Omega)}$  with respect to the outer iterations  $n$ .

### 5.4.3. Parallel Algorithm

Finally, we demonstrate the efficiency of the proposed parallel domain decomposition method (see Algorithm 5) when implemented on a multiple processor computer. We compare its performance with the split Bregman algorithm [30], which computes a solution of (5.1) without any domain decomposition. Remark, that other algorithms for computing a solution of (5.1), see for example [6, 9], might be used as well, changing the result maybe qualitatively but not quantitatively. For a fair comparison, in the domain decomposition method we use the split

## I. Subspace Correction Methods for Total Variation Minimization



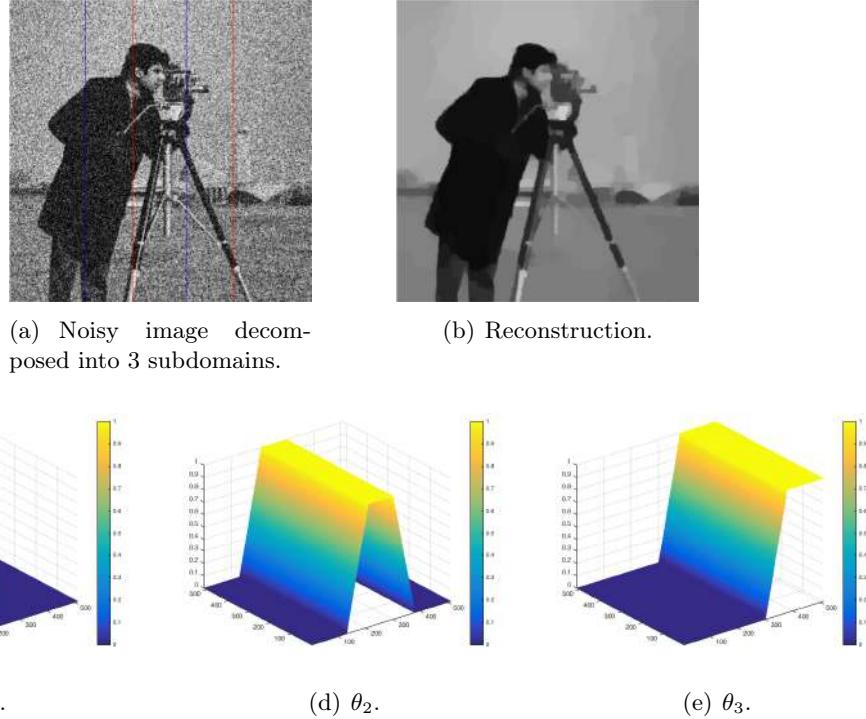
**Figure 5.6:** Reconstruction of an image ( $512 \times 512$  pixels) corrupted by Gaussian white noise with standard deviation  $\sigma = 0.3$  using the regularization parameter  $\alpha = 0.5$ . The noisy image is decomposed into 2 overlapping subdomains with overlap-size  $512 \times 80$  pixels. The blue and red line indicate the interfaces of the overlapping region.

Bregman method as subdomain solver, described in Algorithm 6, and stop it as soon as (5.30) holds for the first time for a given tolerance  $\text{tol} := \text{tol}(n)$ , which depends here on the outer iteration  $n$ . This iteration dependent tolerance seems reasonable to us, since we realized in our numerical tests, that in the first outer iterations the subdomain problems do not need to be solved very accurately, due to the averaging of the current and previous iterates in the update of  $u^n$ , see Algorithm 5. Therefore we set

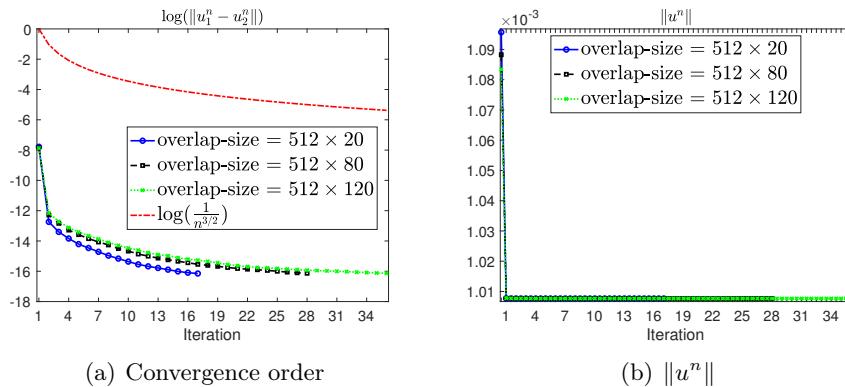
$$\text{tol}(n) = \begin{cases} 10^{-4} & \text{if } 1 \leq n \leq 4 \\ 5 \cdot 10^{-5} & \text{if } 4 < n \leq 7 \\ 10^{-5} & \text{if } 7 < n \leq 50 \\ 8 \cdot 10^{-6} & \text{if } 50 < n \leq 400 \\ 5 \cdot 10^{-6} & \text{if } n > 400 \end{cases},$$

which is chosen empirically and not optimized in any way. We consider partitions of the image domain into  $M = 2, 4, 8$  subdomains and utilize Algorithm 5 to compute a solution of (5.1). For a splitting into 4 and 8 domains we consider a decomposition without and with colouring technique. In case of using the colouring technique the domains are coloured as described in Figure 5.5 leading to  $M_c = 2$ . Since we are comparing the convergence speed of different algorithms, we terminate the algorithms as soon as the energy  $J$  drops below a certain critical

## 5. Overlapping Domain Decomposition for Total Variation Denoising



**Figure 5.7:** Reconstruction of an image ( $512 \times 512$  pixels) corrupted by Gaussian white noise with standard deviation  $\sigma = 0.3$  using the regularization parameter  $\alpha = 0.5$ . The noisy image is decomposed into 3 overlapping subdomains with overlap-size  $512 \times 80$  pixels. The blue and red line indicate the interfaces of the overlapping region.

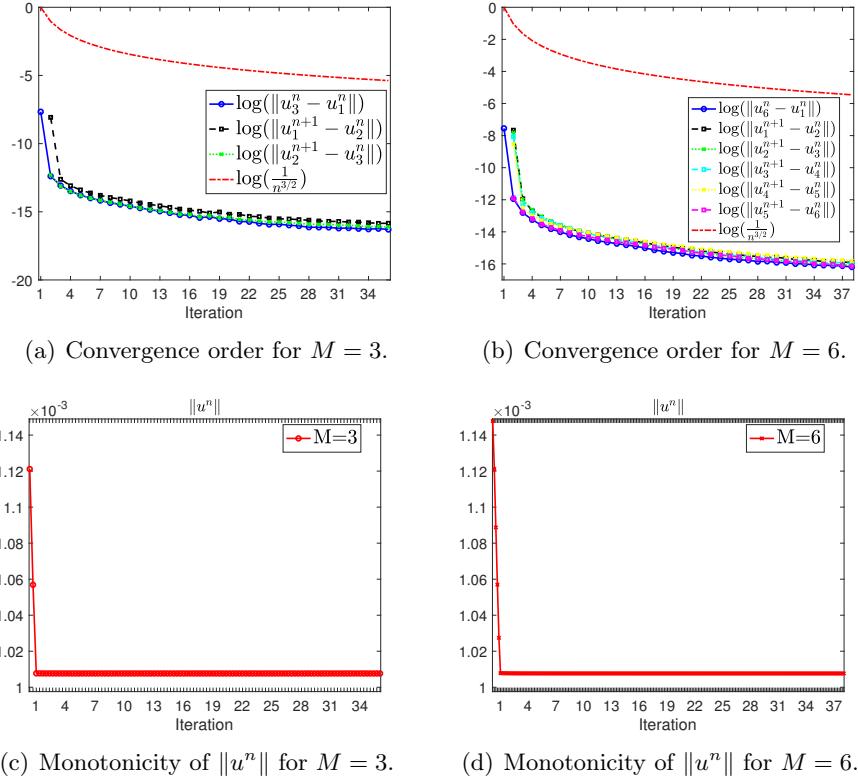


**Figure 5.8:** Performance record of the reconstruction of the image in Figure 5.6 with different overlapping-sizes.

energy  $J^*$  for the first time. This critical energy is obtained empirically by solving the global problem very accurately, so that  $J^*$  is very close to the true minimum.

For our comparison we consider the image in Figure 5.10(a) of size  $1024 \times 1024$  pixels, which has been corrupted by additive Gaussian noise with standard deviation  $\sigma = 0.1$  and zero mean.

## I. Subspace Correction Methods for Total Variation Minimization



**Figure 5.9:** Performance record of the reconstruction of the image in Figure 5.6 with different numbers of subdomains.

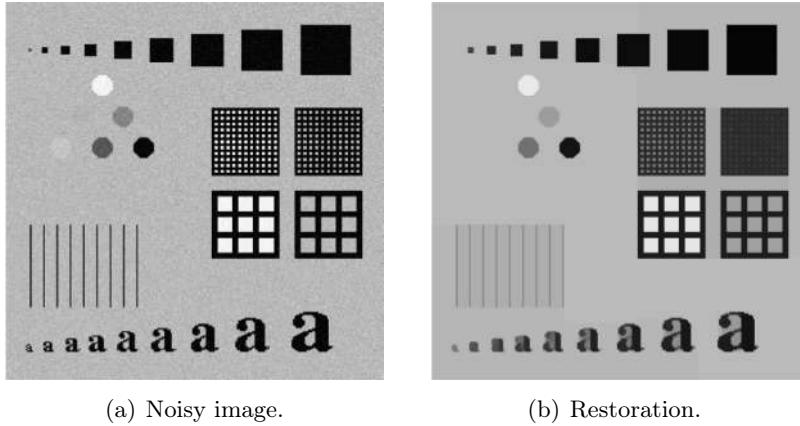
In the parallel domain decomposition algorithm as well as in the split Bregman algorithm we denoise the image by choosing  $\alpha = 1$ . In the parallel domain decomposition algorithm the size of the overlapping region is set to be  $1024 \times 20$  pixels. In Table 5.1 we show for different numbers of subdomains the required time (in seconds) and the number of iterations until the algorithms reached the significant energy  $J^* = 0.018178900879$ . The restored image is shown in Figure 5.10(b). Note, that by domain decomposition, on the one hand, we reduce the dimensionality of the problem, but, on the other hand, in each outer iteration the update  $v_i^{n+1}$  is a weighted sum of the current and previous iterate. This averaging is needed for theoretical reasons, in particular to ensure the convergence to the minimizer of the global problem. The colouring technique reduces the averaging effect, which leads to a faster convergence. It is obvious that with increasing number of subdomains  $M$  this effect becomes more and more visible, see Table 5.1. Additionally, we also have to take the communication time of the processors into account. All these facts sum up to the actual computing time. Hence, we cannot expect a very dramatic decrease in computational time. Nevertheless, we observe from Table 5.1 that the domain decomposition algorithm with splitting into  $M = 2, 4, 8$  subdomains is faster than the split Bregman algorithm computing the solution on the whole domain (1 domain). Thereby, for a decomposition into 8 domains using the colouring technique the best performance with respect to time and iterations is obtained. More precisely, using the parallel domain decomposition method with 8 domains and colouring technique reduces the overall

## 5. Overlapping Domain Decomposition for Total Variation Denoising

	CPU time (s)	No. outer iterations
1 domain	34943.73	9738
2 domains	29228.35	4571
4 domains	22708.67	4470
4 domains with colouring	22205.89	4451
8 domains	21156.57	4253
8 domains with colouring	20481.00	4147

**Table 5.1:** Regularization parameter  $\alpha = 1$ , stopping criterion  $J^* = 0.018178900879$ .

computational time by more than 40% compared with no decomposition.



**Figure 5.10:** Reconstruction of an image of size  $1024 \times 1024$  pixels corrupted by additive Gaussian white noise with  $\sigma = 0.1$ .

## 5.5. Conclusion

We developed convergent overlapping domain decomposition methods for the Rudin-Osher-Fatemi (ROF) problem (5.1) by directly splitting the (primal) problem into respective subdomain problems. In particular, we proved convergence of our proposed splitting methods to a minimizer of the global problem in a continuous setting. We presented two different ways in solving the subdomain problems leading to two similar but still different implementations.

Due to the shape of our subdomain problems, the presented domain decomposition methods are easily applicable to optimization problems with a spatially varying regularization weight, i.e. to problems of the form

$$\min_{u \in L^2(\Omega)} \frac{1}{2} \|u - g\|_{L^2(\Omega)}^2 + \int_{\Omega} \alpha |Du|$$

where  $\alpha : \Omega \rightarrow \mathbb{R}^+$  is a continuous and bounded function, cf. [39, 41]. This type of problem is gaining recently more and more attention, since it allows to penalize homogeneous regions strongly, while in image parts with fine details only little regularization is performed.

## I. Subspace Correction Methods for Total Variation Minimization

### 5.6. References

- [1] R. Acar and C. R. Vogel. Analysis of bounded variation penalty methods for ill-posed problems. *Inverse Problems*, 10(6):1217–1229, 1994.
- [2] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE Transactions on Signal Processing*, 45(4):913–917, 1997.
- [3] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, 2000.
- [4] G. Aubert and J.-F. Aujol. A variational approach to removing multiplicative noise. *SIAM Journal on Applied Mathematics*, 68(4):925–946, 2008.
- [5] S. Bartels. *Numerical Methods for Nonlinear Partial Differential Equations*, volume 14. Springer, 2015.
- [6] M. Burger, A. Sawatzky, and G. Steidl. First order algorithms in variational image processing. In *Splitting Methods in Communication, Imaging, Science, and Engineering*, pages 345–407. Springer, 2016.
- [7] L. Calatroni, J. C. D. L. Reyes, and C.-B. Schönlieb. Infimal convolution of data discrepancies for mixed noise removal. *SIAM Journal on Imaging Sciences*, 10(3):1196–1233, 2017.
- [8] C. Carstensen. Domain decomposition for a non-smooth convex minimization problem and its application to plasticity. *Numerical linear algebra with applications*, 4(3):177–190, 1997.
- [9] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock. An introduction to total variation for image analysis. *Theoretical foundations and numerical methods for sparse recovery*, 9:263–340, 2010.
- [10] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [11] T. F. Chan and T. P. Mathew. Domain decomposition algorithms. *Acta Numerica*, 3:61–143, 1994.
- [12] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001.
- [13] H. Chang, X.-C. Tai, L.-L. Wang, and D. Yang. Convergence rate of overlapping domain decomposition methods for the Rudin–Osher–Fatemi model based on a dual formulation. *SIAM Journal on Imaging Sciences*, 8(1):564–591, 2015.
- [14] K. Chen and X.-C. Tai. A nonlinear multigrid method for total variation minimization from image restoration. *Journal of Scientific Computing*, 33(2):115–138, 2007.
- [15] N. Chumchob, K. Chen, and C. Brito-Loeza. A new variational model for removal of combined additive and multiplicative noise and a fast algorithm for its numerical approximation. *International Journal of Computer Mathematics*, 90(1):140–161, 2013.
- [16] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.
- [17] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure Applied Mathematics*, 57(11):1413–1457, 2004.
- [18] I. Daubechies, G. Teschke, and L. Vese. Iteratively solving linear inverse problems under general convex constraints. *Inverse Problems and Imaging*, 1(1):29–46, 2007.

## 5. Overlapping Domain Decomposition for Total Variation Denoising

- [19] J. C. De los Reyes and C.-B. Schönlieb. Image denoising: Learning the noise model via nonsmooth PDE-constrained optimization. *Inverse Problems and Imaging*, 7(4), 2013.
- [20] Y. Duan, H. Chang, and X.-C. Tai. Convergent non-overlapping domain decomposition methods for variational image segmentation. *Journal of Scientific Computing*, 69(2):532–555, 2016.
- [21] Y. Duan and X.-C. Tai. Domain decomposition methods with graph cuts algorithms for total variation minimization. *Advances in Computational Mathematics*, 36(2):175–199, 2012.
- [22] I. Ekeland and R. Témam. *Convex Analysis and Variational Problems*, volume 28 of *Classics in Applied Mathematics*. SIAM, Philadelphia, PA, 1999.
- [23] M. Fornasier. Domain decomposition methods for linear inverse problems with sparsity constraints. *Inverse Problems*, 23(6):2505, 2007.
- [24] M. Fornasier, Y. Kim, A. Langer, and C.-B. Schönlieb. Wavelet decomposition method for  $L_2/\text{TV}$ -image deblurring. *SIAM Journal on Imaging Sciences*, 5(3):857–885, 2012.
- [25] M. Fornasier, A. Langer, and C.-B. Schönlieb. Domain decomposition methods for compressed sensing. In *Proceedings of the International Conference of SampTA09*, Marseilles, 2009.
- [26] M. Fornasier, A. Langer, and C.-B. Schönlieb. A convergent overlapping domain decomposition method for total variation minimization. *Numerische Mathematik*, 116(4):645–685, 2010.
- [27] M. Fornasier and C.-B. Schönlieb. Subspace correction methods for total variation and  $l_1$ -minimization. *SIAM Journal on Numerical Analysis*, 47(5):3397–3428, 2009.
- [28] P. Getreuer, M. Tong, and L. A. Vese. A variational model for the restoration of MR images corrupted by blur and Rician noise. In *Advances in Visual Computing*, pages 686–698. Springer, 2011.
- [29] E. Giusti. *Minimal Surfaces and Functions of Bounded Variation*, volume 80 of *Monographs in Mathematics*. Birkhäuser Verlag, Basel, 1984.
- [30] T. Goldstein and S. Osher. The split Bregman method for  $L_1$ -regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.
- [31] M. Hintermüller and K. Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM Journal on Applied Mathematics*, 64(4):1311–1333, 2004.
- [32] M. Hintermüller and A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1/L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.
- [33] M. Hintermüller and A. Langer. Surrogate functional based subspace correction methods for image processing. In *Domain Decomposition Methods in Science and Engineering XXI*, pages 829–837. Springer, 2014.
- [34] M. Hintermüller and A. Langer. Non-overlapping domain decomposition methods for dual total variation based image denoising. *Journal of Scientific Computing*, 62(2):456–481, 2015.
- [35] M. Hintermüller and C. Rautenberg. On the density of classes of closed convex sets with pointwise constraints in Sobolev spaces. *Journal of Mathematical Analysis and Applications*, 426(1):585–593, 2015.
- [36] M. Hintermüller and C. N. Rautenberg. Optimal selection of the regularization function in a weighted total variation model. Part I: Modelling and theory. *Journal of Mathematical Imaging and Vision*, 59(3):498–514, 2017.
- [37] M. Hintermüller, C. N. Rautenberg, T. Wu, and A. Langer. Optimal selection of the regularization function in a weighted total variation model. Part II: Algorithm, its analysis and numerical tests. *Journal of Mathematical Imaging and Vision*, 59(3):515–533, 2017.

## I. Subspace Correction Methods for Total Variation Minimization

- [38] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.
- [39] A. Langer. Automated parameter selection for total variation minimization in image restoration. *Journal of Mathematical Imaging and Vision*, 57(2):239–268, 2017.
- [40] A. Langer. Automated parameter selection in the  $L^1$ - $L^2$ -TV model for removing Gaussian plus impulse noise. *Inverse Problems*, 33(7):074002, 2017.
- [41] A. Langer. Investigating the influence of box-constraints on the solution of a total variation model via an efficient primal-dual method. *Journal of Imaging*, 4:12, 2018.
- [42] A. Langer. Locally adaptive total variation for removing mixed Gaussian-impulse noise. *International Journal of Computer Mathematics*, pages 1–19, 2018.
- [43] A. Langer, S. Osher, and C.-B. Schönlieb. Bregmanized domain decomposition for image restoration. *Journal of Scientific Computing*, 54(2-3):549–576, 2013.
- [44] T. Le, R. Chartrand, and T. J. Asaki. A variational approach to reconstructing images corrupted by Poisson noise. *Journal of Mathematical Imaging and Vision*, 27(3):257–263, 2007.
- [45] C.-O. Lee and C. Nam. Primal domain decomposition methods for the total variation minimization, based on dual decomposition. *SIAM Journal on Scientific Computing*, 39(2):B403–B423, 2017.
- [46] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40(3):965–994, 2002.
- [47] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1-2):99–120, 2004.
- [48] A. Quarteroni and A. Valli. *Domain decomposition methods for partial differential equations*. Number CMCS-BOOK-2009-019. Oxford University Press, 1999.
- [49] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.
- [50] C.-B. Schönlieb. Total variation minimization with an  $H^{-1}$  constraint. *CRM Series 9, Singularities in Nonlinear Evolution Phenomena and Applications proceedings, Scuola Normale Superiore Pisa*, pages 201–232, 2009.
- [51] X.-C. Tai and P. Tseng. Convergence rate analysis of an asynchronous space decomposition method for convex minimization. *Mathematics of Computation*, 71(239):1105–1135, 2002.
- [52] X.-C. Tai and J. Xu. Global and uniform convergence of subspace correction methods for some convex optimization problems. *Mathematics of Computation*, 71(237):105–124, 2002.
- [53] A. Toselli and O. B. Widlund. *Domain decomposition methods: algorithms and theory*, volume 34. Springer, 2005.
- [54] P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117(1):387–423, 2009.
- [55] C. Vonesch and M. Unser. A fast multilevel algorithm for wavelet-regularized image restoration. *IEEE Transactions on Image Processing*, 18(3):509–523, 2009.
- [56] Y. Wang, W. Yin, and Y. Zhang. A fast algorithm for image deblurring with total variation regularization. *CAAM Technical Report TR07-10*, 2007.
- [57] J. Warga. Minimizing certain convex functions. *Journal of the Society for Industrial and Applied Mathematics*, 11:588–593, 1963.
- [58] J. Xu, X.-C. Tai, and L.-L. Wang. A two-level domain decomposition method for image restoration. *Inverse Problems and Imaging*, 4(3):523–545, 2010.

## Part II.

# Parameter Selection Methods for Total Variation Models



## 6. Introduction and Overview

It is obvious, that the solutions of problems (1.1) and (1.2) significantly depend on the parameter  $\lambda$ . Hence, the proper choice of  $\lambda$  is extremely important, see Figure 7.1. In particular, too large parameters lead to an over-smoothed reconstruction such that not only noise but also details of the image are eliminated. On the contrary, small parameters  $\lambda$  allow to preserve fine details in the image but also retain noise in homogeneous regions. Hence, in order to obtain a good image reconstruction,  $\lambda$  has to be chosen in such a way that a good compromise between the aforementioned effects is made.

There exists a variety of different methods, which automatically choose regularization parameters. We would like to mention some of them, by stressing that the list is by far not exhaustive.

- (i) *The L-curve method* [23, 24]: This method is based on a parametric plot of the norm of the regularized solution versus the norm of the respective residual. That is, let for example  $u_\lambda$  be a regularized solution of problem (1.1), then one would plot the solution norm  $\|u_\lambda\|_{BV(\Omega)}$  versus the residual norm  $\|Tu_\lambda - g\|_{L^2(\Omega)}$  in a logarithmic scale. The resulting graph generates an L-shaped curve. Then the regularization parameter creating the corner point of the curve is chosen, since this point divides the flat and vertical parts of the curve where the solution is dominated by regularization errors and perturbation errors. In order to obtain the so-called “L-curve”, the respective optimization problem needs to be solved plenty of times for different parameters, which renders this method computationally very expensive. Further, it is shown, that the L-curve method alone should not be the basis of a convergent parameter choice strategy, since it fails to converge to the original image if the noise level tends to zero [17, 22, 46].
- (ii) *The generalized cross-validation method* [21, 30, 31, 38]: As the L-curve method the generalized cross-validation method is usually used to estimate the regularization parameter if no information about the noise level is known. The method was originally introduced for inverse problems where the solution depends linearly on the observed data. However, due to the nonlinearity of the total variation this method cannot be directly applied to problems (1.1) and (1.2). Nevertheless, the utilization of a variable-splitting method and an alternating minimization scheme in [30] allows to use the generalized cross-validation technique to estimate the regularization parameter for the  $L^2$ -TV model and the  $L^1$ -TV model. Moreover, there exist extensions of the generalized cross-validation method in case of nonlinearity [38] and references therein, but they are computationally rather expensive.
- (iii) *The unbiased predictive risk estimator method* (UPRE) [34, 31]: The UPRE is based on a statistical estimator of the mean squared norm of the predictive error and only valid if the regularized solution depends linearly on the data [46], which is not the case for problems (1.1) and (1.2). An extension to the  $L^2$ -TV model is considered in [31].

## II. Parameter Selection Methods for Total Variation Models

- (iv) *The Stein unbiased risk estimator method* (SURE) [6, 15, 40] and its generalizations [13, 16, 20]: The SURE provides an unbiased estimate of the mean-squared error between the noiseless image and the regularized solution. While the original method is restricted to the case of denoising images contaminated by additive white Gaussian noise, its generalization, named GSURE (generalized SURE), is developed to deal with a more diverse set of inverse problems, e.g., image deblurring and image zooming [16, 20].
- (v) *The discrepancy principle* [36]: The idea of this method is to choose the regularization parameter  $\lambda$  such that given noisy data  $g$  and given noise level  $\delta$ , i.e.,  $\|T\hat{u} - g\| \leq \delta$ , the residual  $\|Tu_\lambda - g\| \leq \tau\delta$ , where  $\tau > 1$  is a constant independent of  $\delta$  and  $g$ . Here  $\|\cdot\|$  denotes a suitable norm and  $u_\lambda$  is a regularized solution with regularization parameter  $\lambda$ . This method seems to be most applied to select parameters for problems (1.1) and (1.2), see [5, 9, 25, 37, 47, 48].
- (vi) *Learning-based methods* [8, 12, 28]: Given a training set of pairs  $(g_k, \hat{u}_k)$  for  $k = 1, 2, \dots, N \in \mathbb{N}$ , where  $g_k$  denotes the noisy observation and  $\hat{u}_k$  represents the underlying original data. Then in [8, 12, 28] based on a bilevel optimization approach suitable regularization parameters are learned for image denoising. Thereby a bilevel optimization problem of the form

$$\min_{u, \lambda} F(u, \lambda) \quad (6.1)$$

$$\text{s.t. } u \in \arg \min_u J(u, \lambda) \quad (6.2)$$

is used, where the lower level problem (6.2) is given by the variation model, e.g., problem (1.1), and the higher level problem (6.1) consists of an error function, which measures the difference between the ground truth  $\hat{u}_k$  and a solution of the lower level problem.

The choice of the parameter selection method depends on the available information of the problem of interest. If no training set is given but the noisy image and the respective noise level is known, then one may use the UPRE, SURE or discrepancy principle. The most popular one of these three seems to be the discrepancy principle, which might be due to the fact that it allows for a pure variational approach for computing an appropriate parameter. This is also the reason why we concentrate in thesis on this method. Moreover, it is worth to mention that for problems (1.1), (1.2), and (1.3) it is not yet fully understood how to choose “optimal” parameters automatically. This might be a reason why most commonly parameters in imaging applications are chosen by *trial-and-error*. This is usually either costly, since the considered problem has to be solved a lot of times, or inaccurate, if only for a few parameters the respective solution is computed. In this part of the thesis we develop new methods, which fully automatically compute suitable parameters for total variation minimization problems and are based on the discrepancy principle.

Applying the discrepancy principle on the  $L^2$ -TV model or the  $L^1$ -TV model, in order to find a suitable parameter  $\lambda$ , we can formulate the image reconstruction problem as

$$\min_{u \in BV(\Omega)} \int_{\Omega} |Du| \quad \text{s.t. } \mathcal{H}_{\tau}(Tu; g) = \mathcal{B}_{\tau}, \quad (6.3)$$

where  $\mathcal{H}_{\tau}(u; g) := \|u - g\|_{L^{\tau}(\Omega)}^{\tau}$ ,  $\mathcal{B}_{\tau} := \frac{\nu_{\tau}}{\tau} |\Omega|$  with  $\nu_{\tau} > 0$  being a constant independent of the underlying noise,  $\tau = 1, 2$ , and  $|\Omega|$  denoting the volume of  $\Omega$ . In order to determine the value

## 6. Introduction and Overview

$\nu_\tau$ , allowing to solve the considered problem, knowledge about the noise level is necessary. This may mean, that in a first step the noise level has to be estimated before the discrepancy principle can be applied. Nevertheless, in general it seems easier to estimate the noise level than the regularization parameter [9].

The constrained optimisation problem (6.3) is naturally linked to the unconstrained problem (1.1) respectively (1.2). This link has to be understood in the sense, that there exist  $\lambda \geq 0$  such that the solution of problem (1.1) respectively (1.2) also solves problem (6.3), meaning that the respective optimisation problems are equivalent. Due to this connection different methods based on the discrepancy principle and formulation (6.3) with  $\tau = 2$  are presented in the literature, see for example [5, 9, 25, 48]. In contrast, only very little attention has been given to the case  $\tau = 1$ , see for example [37, 47].

The structure of this part is as follows. In Chapter 7, see [AL5], we present new methods, which automatically determine the regularization parameter  $\lambda$  in (1.1) or (1.2) and are based on formulation (6.3). Our approach is motivated by the parameter choice method in [9], which is only developed for the  $L^2$ -TV model with  $T = I$ . We adapt this approach for the  $L^2$ -TV model and the  $L^1$ -TV model for general linear and bounded operators  $T$  [AL5]. By using an appropriate update for  $\lambda$ , which differs from the one in [9], we prove in [AL5] theoretically and numerically, that our method indeed converges towards the desired regularization parameter. Besides general applicability of our method, it possesses even advantages over the algorithm in [9] for the case  $\tau = 2$  and  $T = I$ . In particular, in the worst case our method needs as many iterations as the method from [9], while in practice our method usually needs less iterations. This behaviour is not only numerically but also theoretically confirmed.

We remark, that a scalar regularization parameter may not be the optimal choice for every image reconstruction problem, since images usually possess large homogeneous regions as well as parts with a lot of details. Hence, it seems better to choose  $\lambda$  large in homogeneous regions, in order to significantly reduce noise there, and relatively small in parts with fine details, to retain them. This motivated researches to consider multiscale total variation models with spatially varying parameters initially suggested in [39]. The multiscale versions of problems (1.1) and (1.2) read

$$\min_{u \in BV(\Omega)} \mathcal{H}_\tau(u; g) + \int_{\Omega} \lambda(x) |Du| \quad (6.4)$$

for  $\tau = 1, 2$ , where  $\lambda : \Omega \rightarrow \mathbb{R}^+$  is a bounded and continuous function [2, 26]. Note, that instead of problems (1.1) and (1.2) also the equivalent minimization of  $\alpha \mathcal{H}_\tau(Tu; g) + \int_{\Omega} |Du|$  with  $\alpha = \frac{1}{\lambda}$  may be studied, leading to the multiscale version

$$\min_{u \in BV(\Omega)} \int_{\Omega} \alpha(x) |Tu - g|^\tau dx + \int_{\Omega} |Du|, \quad (6.5)$$

where  $\alpha : \Omega \rightarrow \mathbb{R}^+$  is a function and  $\tau \in \{1, 2\}$ .

The influence of the scale of an image feature on the choice of  $\lambda$  is studied in [42] and its observations are used in [41] to construct an updating scheme of  $\lambda$ . For problem (6.5) in [4] a piecewise constant function  $\alpha$ , where the pieces are defined by a partitioning of the image due to a pre-segmentation, is determined. In particular, for each segment a scalar  $\alpha_i$ ,  $i = 1, \dots, \#\text{pieces}$  is computed by Uzawa's method [11]. Based on statistics of local constraints in [1, 14, 19] for problem (6.5) automated update rules for  $\alpha$  are proposed. In [19] a two-level approach for variational denoising is considered, where in the first-level noise and relevant texture are isolated in order to compute local constraints based on local variance estimation.

## II. Parameter Selection Methods for Total Variation Models

In the second-level a gradient descent method and an update formula for  $\alpha(x)$  derived from the Euler–Lagrange equation is utilized. An adaptation of this approach to multiplicative noise can be found in [29]. For convolution type of problems in [1] based on an estimate of the noise variance for each pixel an automatic updating scheme of  $\alpha$  using Uzawa’s method is created. This approach is improved in [14] by determining the fidelity weights due to the Gumbel statistic for the maximum of a finite number of random variables associated with localized image residuals and by incorporating hierarchical image decompositions proposed in [44, 45] to speed up the iterative parameter adjustment process. An adaptation of this approach to a total variation model with  $L^1$  local constraints is studied in [27]. A different approach has been proposed in [43] for image denoising only, where nonlocal means [7] are used to create a nonlocal data fidelity term.

While the methods in [14, 27], based on (6.5) are competitive, the iterative adjustment of  $\alpha$  requires, that  $T$  maps into the image domain  $\Omega$ . This limits the applicability of these approaches. In particular, applications in which the operator  $T$  is a transformation of the image data into some data space disparate of the image space, e.g.,  $T$  being a wavelet or Fourier transform, these methods cannot be used. In Chapter 8, see also [**AL6**, **AL7**], we consider exactly such applications, i.e.,  $T$  being a wavelet or the Fourier transform, and adapt and study the locally adaptive parameter selection strategy proposed in [14] for such non-image data, which are possibly subsampled. In order to deal with this situation, we replace (6.5) for  $\tau = 2$  by a sequence of auxiliary problems. These auxiliary problems are chosen in a way, that (i)  $T$  does not affect  $u$  any longer and hence each auxiliary problem is solved in the spatial domain, and (ii) the sequence of solutions of the auxiliary problems converges towards the minimizer of the original problem. Since the sequence of auxiliary problems is solved in the spatial domain, we are able to choose the regularization parameter locally varying, similar as in [14]. Numerical experiments demonstrate, that this new method is able to generate reconstructions from partial Fourier data, as they occur in magnetic resonance imaging, and from partial wavelet coefficients, which are better than the results of standard methods with globally constant parameters.

Instead of selecting the locally dependent weight  $\alpha$  in problem (6.5) we present in Chapter 7, see also [**AL5**], an automatic selection strategy of locally adaptive weights  $\lambda$  for the problem (6.4). In order to choose an appropriate locally varying  $\lambda$ , such that fine details are preserved and noise in uniform parts is considerably removed, we construct an iterative algorithm based on local constraints leading to the following problem

$$\min_{u \in BV(\Omega)} \int_{\Omega} |Du| \quad \text{s.t.} \quad S(u)(x) \leq B \quad \text{f.a.a. } x \in \Omega, \quad (6.6)$$

where  $S(u)(x)$  denotes the local residual of an approximation  $u$  in  $x \in \Omega$  and  $B$  is a (local) value, usually dependent on the underlying noise, cf.  $\mathcal{B}_\tau$  in (6.3). Then for a given (possibly locally varying)  $\lambda$  and the associated solution  $u_\lambda$  of problem (6.4) we perform iteratively as follows: If  $S(u_\lambda)(x) > B$ , then we assume that details of the image are contained in the local residual and we decrease  $\lambda$  in  $x$ . If  $S(u_\lambda)(x) < B$ , then  $\lambda$  in  $x$  is increased and if  $S(u_\lambda)(x) = B$ , then  $\lambda$  in  $x$  stays unchanged.

Numerical examples show, that our locally adaptive method is able to generate a locally varying  $\lambda$ , whose associated reconstruction, i.e., a minimizer of problem (6.4), is qualitatively better, than the reconstruction obtained with a scalar parameter choice method. A comparison of our method with the ones in [14, 27] demonstrates its competitiveness.

## 6. Introduction and Overview

Moreover, for globally constant  $B$  we show analytically in [AL5], that a solution of the locally constrained optimisation problem (6.6) preserves more details, than a solution of the globally constrained problem (6.3). More precisely, let  $u_s$  be a solution of (6.3) and  $u_l$  be a solution of the locally constrained problem, then we have  $\int_{\Omega} |Du_s| \leq \int_{\Omega} |Du_l|$ , see [AL5, Proposition 4.2].

In case the dynamic range  $[c_{\min}, c_{\max}]$  of the original image  $\hat{u}$  is known, i.e.,  $c_{\min} \leq \hat{u}(x) \leq c_{\max}$  f.a.a.  $x \in \Omega$ , it seems very reasonable to use this information in the reconstruction process. In particular, incorporating this knowledge into the  $L^2$ -TV model leads to

$$\min_{u \in BV(\Omega) \cap C} \|Tu - g\|_{L^2(\Omega)}^2 + \lambda \int_{\Omega} |Du| \quad (6.7)$$

where  $C := \{u \in L^2(\Omega) : c_{\min} \leq u(x) \leq c_{\max} \text{ f.a.a. } x \in \Omega\}$  is the solution space containing the dynamic range as box-constraints.

In connection with image reconstruction in [3, 10, 33, 35] box-constraints have been already considered. In [35] a functional consisting of an  $L^2$  data term and a Tikonov-like regularization term together with box-constraints is presented. As this functional is differentiable, a Newton-type method could be used to solve the respective constituted optimization problem. For the minimization of the total variation subject to box-constraints a fast algorithm, called monotone fast iterative shrinkage/thresholding algorithm (MFISTA), is proposed in [3] and its convergence rate is shown. Based on the alternating direction method of multipliers (ADMM) [18] a solver for box-constrained versions of the  $L^2$ -TV and  $L^1$ -TV model are derived in [10]. It is even shown, that this solver converges faster than MFISTA. In [33] a primal-dual algorithm for the  $L^1$ -TV model with box-constraints and for the minimization of the nonlocal total variation subject to box-constraints is presented. In order to achieve a constrained solution, which is positive and bounded from above by some intensity value, in [49] an exponential-type transform is applied to the  $L^2$ -TV model. Recently also the  $L^1$ - $L^2$ -TV model is equipped with a box-constraint in [32] to remove simultaneously Gaussian and impulse noise.

It is obvious, that in general a solution of the  $L^2$ -TV model is not an element of  $C$ . However, if the regularization parameter  $\lambda$  in (1.1) is large enough, than clearly the minimizer of the  $L^2$ -TV model is indeed in  $C$ . In such a case box-constraints in (6.7) do not have any influence on the solution. This poses the question, whether an optimally chosen parameter anyway yields a solution in  $C$ .

In order to answer this question we determine in Chapter 9, published in [AL8], numerically nearly optimal values for  $\lambda$  of the  $L^2$ -TV model, the constrained problem (6.7), and their locally adaptive versions, cf. problem (6.4). We compare the respective reconstructions and observe, that in most applications it is more a question of a suitable choice of the regularization parameter, than incorporating box-constraints, resulting from the dynamic range. For a special case we even show analytically, that the solution of the optimisation problem without box-constraints indeed is an element of  $C$  and hence a box-constraint does not have any influence on the solution. However, in the application of reconstructing sub-sampled Radon-data, as it occurs in computerized tomography, we observe, that an additional nonnegativity constraint significantly improves the restoration. Note, that the dynamic range of an image is naturally bounded from below by zero. However, because of the Radon-transform some values in the optimization process may become significantly negative, such that these values have an adverse influence on the reconstruction quality. Hence, in this particular application an additional constraint seems reasonable.

## II. Parameter Selection Methods for Total Variation Models

Based on the observation, that in most applications the solution is affected by the additional box-constraint, it seems suitable to construct a parameter choice rule based on the image intensity range. Of course, this seems only then useful, if no information about the noise can be obtained but the image intensity range is known. For such a situation we construct in [AL8] an automatic parameter selection method. We observe in our numerical experiments, that the reconstructions obtained via this method are similar but still slightly inferior with respect to image quality measures to the ones generated by the parameter selection method in [AL5], where the noise level is known. Hence, this method is an alternative to standard parameter selection methods when instead of the noise level the image intensity range is known.

Instead of assuming, that the observed image is only corrupted by one type of noise, for instance leading to the  $L^2$ -TV model in case of Gaussian noise or to the  $L^1$ -TV model in case of impulse noise, a much more realistic approach supposes, that several different types of noise appear in an observation. In this vein, we proposed in Chapter 2, cf. [AL1], the  $L^1$ - $L^2$ -TV model, see problem (1.3), which is able to eliminate mixed Gaussian-impulse noise simultaneously. Similar to the  $L^2$ -TV and  $L^1$ -TV model, the minimizer of the  $L^1$ - $L^2$ -TV model depends significantly on the choice of the parameters  $\alpha_1$  and  $\alpha_2$ . In [32] it is suggested to choose these parameters according to the standard deviation  $\sigma$  of the Gaussian noise and an energy of the impulse noise, i.e.

$$\alpha_1 = \frac{E_I}{E_I + \sigma^2} \quad \text{and} \quad \alpha_2 = \frac{\sigma^2}{E_I + \sigma^2}, \quad (6.8)$$

where  $E_I = \frac{s_1+s_2}{2}$  for salt-and-pepper noise and  $E_I = \frac{s}{3}$  for random-valued impulse noise. Although the numerical experiments look promising, it is neither clear nor analytically justified in which sense the formulas in (6.8) provide suitable parameters for the  $L^1$ - $L^2$ -TV model.

Based on a stochastic characterisation of the mixed Gaussian-impulse noise, we present in Chapter 10, published in [AL9], a fully automatic parameter selection strategy to determine the parameters  $\alpha_1$  and  $\alpha_2$  in the  $L^1$ - $L^2$ -TV model. This approach is based on the discrepancy principle. Therefore, we relate the  $L^1$ - $L^2$ -TV model to a respective constrained optimisation problem. More precisely, we consider the constrained problem

$$\min_{u \in BV(\Omega)} \int_{\Omega} |Du| \quad \text{s.t.} \quad \|T_1 u - g_1\|_{L^1(\Omega)} \leq \nu_1 |\Omega| \quad \text{and} \quad \|T_2 u - g_2\|_{L^2(\Omega)}^2 \leq \nu_2 |\Omega|, \quad (6.9)$$

where  $\nu_1, \nu_2 \geq 0$  are stochastic values dependent on the underlying noise. Under certain assumptions, actually, that there is a feasible  $u \in BV(\Omega)$  and that  $T_i$  does not annihilate constant functions for at least one  $i \in \{1, 2\}$ , the constrained problem (6.9) has a solution [AL9, Theorem 3.2]. Similarly, assuming again that  $T_i$  does not annihilate constant functions for at least one  $i \in \{1, 2\}$ , the existence of minimizers of the  $L^1$ - $L^2$ -TV model (1.3) is also proven [AL9]. Additionally we show, that there are  $\alpha_1$  and  $\alpha_2$ , such that a solution of the unconstrained  $L^1$ - $L^2$ -TV model is equivalent to a solution of the constrained problem (6.9). This theoretical result allows us in [AL9] to construct an automated parameter selection method for  $\alpha_1$  and  $\alpha_2$ . In particular, the proposed method utilizes an interplay between the penalized problem, i.e., the  $L^1$ - $L^2$ -TV model, and the associated constrained problem (6.9) in order to either increase or decrease the parameters  $\alpha_i$ ,  $i = 1, 2$ , in each iteration. Since this update rule generates a monotonic sequence of parameters, we are able to prove the convergence of the proposed method. Numerical experiments demonstrate, that the proposed method selects parameters  $\alpha_1$  and  $\alpha_2$  such that the corresponding restoration is better with respect to some restoration quality measure than the one obtained by the formulas in (6.8).

## 6. Introduction and Overview

As in general, a scalar parameter is not the best choice and work [AL5] is restricted to the  $L^2$ -TV and  $L^1$ -TV models, we consider in Chapter 11, see [AL10], a locally adaptive  $L^1$ - $L^2$ -TV model, i.e.,

$$\min_{u \in BV(\Omega)} \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \int_{\Omega} \lambda(x) |Du|. \quad (6.10)$$

Under the same assumption as in [AL9], the existence of a solution of model (6.10) is guaranteed [AL10]. In order to determine a suitable  $\lambda$ , we consider constraints which are of the form as in (6.6), whereas here  $S(u)(\cdot) := \int_{\Omega} w(\cdot, y) \mathcal{H}(u)(y) dy$  with  $w$  being a normalised localization filter and  $\mathcal{H}(u)(y) := \alpha_1 |T_1 u - g_1|(y) + \alpha_2 |T_2 u - g_2|^2(y)$ . Thereby, we pursue two different approaches regarding the choice of  $B$ . In the first approach the value  $B$  is chosen to be the same for every local constraint, while in the second approach  $B$  is allowed to vary locally, i.e., we adjust  $B$  locally to the noise. An algorithm, similar to the one in [AL5], is constructed in [AL10] for solving problem (6.10). With the help of this algorithm we show numerically in [AL10], that a locally adjusted  $B$  generates much better results than a globally constant  $B$ .

## References

- [1] A. Almansa, C. Ballester, V. Caselles, and G. Haro. A TV based restoration model with local constraints. *Journal of Scientific Computing*, 34(3):209–236, 2008.
- [2] H. Attouch, G. Buttazzo, and G. Michaille. *Variational Analysis in Sobolev and BV Spaces: Applications to PDEs and optimization*. MOS-SIAM Series on Optimization. SIAM, second edition, 2014.
- [3] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18(11):2419–2434, 2009.
- [4] M. Bertalmío, V. Caselles, B. Roug  , and A. Sol  . TV based image restoration with local constraints. *Journal of Scientific Computing*, 19(1-3):95–122, 2003.
- [5] P. Blomgren and T. F. Chan. Modular solvers for image restoration problems using the discrepancy principle. *Numerical Linear Algebra with Applications*, 9(5):347–358, 2002.
- [6] T. Blu and F. Luisier. The SURE-LET approach to image denoising. *IEEE Transactions on Image Processing*, 16(11):2778–2786, 2007.
- [7] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.
- [8] L. Calatroni, C. Chung, J. C. D. L. Reyes, C.-B. Sch  nlieb, and T. Valkonen. Bilevel approaches for learning of variational imaging models. In: *Variational Methods in Imaging and Geometric Control, Radon Series on Computational and Applied Mathematics*, volume 18, pages 252–290, 2016.
- [9] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97, 2004.
- [10] R. H. Chan, M. Tao, and X. Yuan. Constrained total variation deblurring models and fast algorithms based on alternating direction method of multipliers. *SIAM Journal on Imaging Sciences*, 6(1):680–697, 2013.
- [11] P. G. Ciarlet. *Introduction to Numerical Linear Algebra and Optimisation*. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge, 1989.

## II. Parameter Selection Methods for Total Variation Models

- [12] J. C. De los Reyes and C.-B. Schönlieb. Image denoising: Learning the noise model via nonsmooth PDE-constrained optimization. *Inverse Problems and Imaging*, 7(4), 2013.
- [13] C.-A. Deledalle, S. Vaiter, J. Fadili, and G. Peyré. Stein unbiased gradient estimator of the risk (sugar) for multiple parameter selection. *SIAM Journal on Imaging Sciences*, 7(4):2448–2487, 2014.
- [14] Y. Dong, M. Hintermüller, and M. M. Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *Journal of Mathematical Imaging and Vision*, 40(1):82–104, 2011.
- [15] D. L. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224, 1995.
- [16] Y. C. Eldar. Generalized sure for exponential families: Applications to regularization. *IEEE Transactions on Signal Processing*, 57(2):471–481, 2009.
- [17] H. W. Engl and W. Grever. Using the L-curve for determining optimal regularization parameters. *Numerische Mathematik*, 69(1):25–31, 1994.
- [18] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40, 1976.
- [19] G. Gilboa, N. Sochen, and Y. Y. Zeevi. Texture preserving variational denoising using an adaptive fidelity term. In *Proceedings of the 2nd IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision*, 2003.
- [20] R. Giryes, M. Elad, and Y. C. Eldar. The projected GSURE for automatic parameter tuning in iterative shrinkage methods. *Applied and Computational Harmonic Analysis*, 30(3):407–422, 2011.
- [21] G. H. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.
- [22] M. Hanke. Limitations of the  $L$ -curve method in ill-posed problems. *BIT Numerical Mathematics*, 36(2):287–301, 1996.
- [23] P. C. Hansen. Analysis of discrete ill-posed problems by means of the  $L$ -curve. *SIAM Review*, 34(4):561–580, 1992.
- [24] P. C. Hansen and D. P. O’Leary. The use of the  $L$ -curve in the regularization of discrete ill-posed problems. *SIAM Journal on Scientific Computing*, 14(6):1487–1503, 1993.
- [25] C. He, C. Hu, W. Zhang, and B. Shi. A fast adaptive parameter estimation for total variation image restoration. *IEEE Transactions on Image Processing*, 23(12):4954–4967, 2014.
- [26] M. Hintermüller and C. N. Rautenberg. Optimal selection of the regularization function in a weighted total variation model. Part I: Modelling and theory. *Journal of Mathematical Imaging and Vision*, pages 1–17, 2017.
- [27] M. Hintermüller and M. M. Rincon-Camacho. Expected absolute value estimators for a spatially adapted regularization parameter choice rule in  $L^1$ -TV-based image restoration. *Inverse Problems*, 26(8):085005, 30, 2010.
- [28] K. Kunisch and T. Pock. A bilevel optimization approach for parameter learning in variational models. *SIAM Journal on Imaging Sciences*, 6(2):938–983, 2013.
- [29] F. Li, M. K. Ng, and C. Shen. Multiplicative noise removal with spatially varying regularization parameters. *SIAM Journal on Imaging Sciences*, 3(1):1–20, 2010.
- [30] H. Liao, F. Li, and M. K. Ng. Selection of regularization parameter in total variation image restoration. *Journal of the Optical Society of America A*, 26(11):2311–2320, 2009.

## 6. Introduction and Overview

- [31] Y. Lin, B. Wohlberg, and H. Guo. UPRE method for total variation parameter selection. *Signal Processing*, 90(8):2546–2551, 2010.
- [32] R. W. Liu, L. Shi, S. C. H. Yu, and D. Wang. Box-constrained second-order total generalized variation minimization with a combined  $L^{1,2}$  data-fidelity term for image reconstruction. *Journal of Electronic Imaging*, 24(3):033026, 2015.
- [33] L. Ma, M. Ng, J. Yu, and T. Zeng. Efficient box-constrained TV-type- $l^1$  algorithms for restoring images with impulse noise. *Journal of Computational Mathematics*, 31:249–270, 2013.
- [34] C. L. Mallows. Some comments on  $C_P$ . *Technometrics*, 15(4):661–675, 1973.
- [35] B. Morini, M. Porcelli, and R. H. Chan. A reduced Newton method for constrained linear least-squares problems. *Journal of Computational and Applied Mathematics*, 233(9):2200–2212, 2010.
- [36] V. A. Morozov. *Methods for Solving Incorrectly Posed Problems*. Springer-Verlag, New York, 1984. Translated from Russian by A. B. Aries.
- [37] M. K. Ng, P. Weiss, and X. Yuan. Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods. *SIAM Journal on Scientific Computing*, 32(5):2710–2736, 2010.
- [38] S. Ramani, Z. Liu, J. Rosen, J. Nielsen, and J. A. Fessler. Regularization parameter selection for nonlinear iterative image restoration and MRI reconstruction using GCV and SURE-based methods. *IEEE Transactions on Image Processing*, 21(8):3659–3672, 2012.
- [39] L. I. Rudin and S. Osher. Total variation based image restoration with free local constraints. In *Proceedings of 1st International Conference on Image Processing (ICIP)*, volume 1, pages 31–35. IEEE, 1994.
- [40] C. M. Stein. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, 9(6):1135–1151, 1981.
- [41] D. M. Strong, P. Blomgren, and T. F. Chan. Spatially adaptive local-feature-driven total variation minimizing image restoration. In *Optical Science, Engineering and Instrumentation'97*, pages 222–233. International Society for Optics and Photonics, 1997.
- [42] D. M. Strong and T. F. Chan. Spatially and scale adaptive total variation based regularization and anisotropic diffusion in image processing. In *Division in Image Processing, UCLA Mathematics Department CAM Report*. Citeseer, 1996.
- [43] C. Sutour, C.-A. Deledalle, and J.-F. Aujol. Adaptive regularization of the nl-means: Application to image and video denoising. *IEEE Transactions on Image Processing*, 23(8):3506–3521, 2014.
- [44] E. Tadmor, S. Nezzar, and L. Vese. A multiscale image representation using hierarchical  $(BV, L^2)$  decompositions. *Multiscale Modeling & Simulation*, 2(4):554–579, 2004.
- [45] E. Tadmor, S. Nezzar, and L. Vese. Multiscale hierarchical decomposition of images with applications to deblurring, denoising and segmentation. *Communications Mathematical Sciences*, 6(2):281–307, 2008.
- [46] C. R. Vogel. *Computational Methods for Inverse Problems*. SIAM, 2002.
- [47] P. Weiss, L. Blanc-Féraud, and G. Aubert. Efficient schemes for total variation minimization under constraints in image processing. *SIAM Journal on Scientific Computing*, 31(3):2047–2080, 2009.
- [48] Y.-W. Wen and R. H. Chan. Parameter selection for total-variation-based image restoration using discrepancy principle. *IEEE Transactions on Image Processing*, 21(4):1770–1781, 2012.
- [49] B. M. Williams, K. Chen, and S. P. Harding. A new constrained total variational deblurring model and its fast algorithm. *Numerical Algorithms*, 69(2):415–441, 2015.



# 7. Automated Parameter Selection for Total Variation Minimization in Image Restoration

**Bibliographic Note:** The content of this chapter is published in the article [AL5]: A. Langer. Automated parameter selection for total variation minimization in image restoration. *Journal of Mathematical Imaging and Vision*, 57(2):230–268, 2017.

**Summary:** Algorithms for automatically selecting a scalar or locally varying regularization parameter for total variation models with an  $L^\tau$ -data fidelity term,  $\tau \in \{1, 2\}$ , are presented. The automated selection of the regularization parameter is based on the discrepancy principle, whereby in each iteration a total variation model has to be minimized. In the case of a locally varying parameter this amounts to solve a multiscale total variation minimization problem. For solving the constituted multiscale total variation model convergent first and second order methods are introduced and analyzed. Numerical experiments for image denoising and image deblurring show the efficiency, the competitiveness, and the performance of the proposed fully automated scalar and locally varying parameter selection algorithms.

## 7.1. Introduction

Observed images are often contaminated by noise and may be additionally distorted by some measurement device. Then the obtained data  $g$  can be described as

$$g = \mathcal{N}(T\hat{u}),$$

where  $\hat{u}$  is the unknown original image,  $T$  is a linear bounded operator modeling the image-formation device, and  $\mathcal{N}$  represents noise. In this paper, we consider images which are contaminated either by white Gaussian noise or impulse noise. While for white Gaussian noise the degraded image  $g$  is obtained as

$$g = T\hat{u} + \eta,$$

where the noise  $\eta$  is oscillatory with zero mean and standard deviation  $\sigma$ , there are two main models for impulse noise, that are widely used in a variety of applications, namely salt-and-pepper noise and random-valued impulse noise. We assume that  $T\hat{u}$  is in the dynamic range  $[0, 1]$ , i.e.,  $0 \leq T\hat{u} \leq 1$ , then in the presence of salt-and-pepper noise the observation  $g$  is given by

$$g(x) = \begin{cases} 0 & \text{with probability } r_1 \in [0, 1), \\ 1 & \text{with probability } r_2 \in [0, 1), \\ T\hat{u}(x) & \text{with probability } 1 - r_1 - r_2, \end{cases} \quad (7.1)$$

## II. Parameter Selection Methods for Total Variation Models

with  $1 - r_1 - r_2 > 0$ . If the image is contaminated by random-valued impulse noise, then  $g$  is described as

$$g(x) = \begin{cases} \rho & \text{with probability } r \in [0, 1), \\ T\hat{u}(x) & \text{with probability } 1 - r, \end{cases} \quad (7.2)$$

where  $\rho$  is a uniformly distributed random variable in the image intensity range  $[0, 1]$ .

The recovery of  $\hat{u}$  from the given degraded image  $g$  is an ill-posed inverse problem and thus regularization techniques are required to restore the unknown image [41]. A good approximation of  $\hat{u}$  may be obtained by solving a minimization problem of the type

$$\min_u \mathcal{H}(u; g) + \alpha \mathcal{R}(u), \quad (7.3)$$

where  $\mathcal{H}(\cdot; g)$  represents a data fidelity term, which enforces the consistency between the recovered and measured image,  $\mathcal{R}$  is an appropriate filter or regularization term, which prevents over-fitting, and  $\alpha > 0$  is a regularization parameter weighting the importance of the two terms. We aim at reconstructions in which edges and discontinuities are preserved. For this purpose we use the total variation as a regularization term, first proposed in [81] for image denoising. Hence, here and in the remaining of the paper we choose  $\mathcal{R}(u) = \int_{\Omega} |Du|$ , where  $\int_{\Omega} |Du|$  denotes the total variation of  $u$  in  $\Omega$ ; see [3, 46] for more details. However, we note that other regularization terms, such as the total generalized variation [13], the nonlocal total variation [60], the Mumford-Shah regularizer [71], or higher order regularizers (see e.g. [77] and references therein) might be used as well.

### 7.1.1. Choice of the Fidelity Term

The choice of  $\mathcal{H}$  typically depends on the type of noise contamination. For images corrupted by Gaussian noise a quadratic  $L^2$ -data fidelity term is typically chosen and has been successfully used; see for example [18, 19, 20, 24, 28, 29, 30, 32, 35, 47, 72, 76, 91, 93]. In this approach, which we refer to as the  $L^2$ -TV model, the image  $\hat{u}$  is recovered from the observed data  $g$  by solving

$$\min_{u \in BV(\Omega)} \frac{1}{2} \|Tu - g\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} |Du|, \quad (7.4)$$

where  $BV(\Omega)$  denotes the space of functions with bounded variation, i.e.,  $u \in BV(\Omega)$  if and only if  $u \in L^1(\Omega)$  and  $\int_{\Omega} |Du| < \infty$ . In the presence of impulse noise, e.g., salt-and-pepper noise or random-valued impulse noise, the above model usually does not yield a satisfactory restoration. In this context, a more successful approach, suggested in [1, 74, 75], uses a nonsmooth  $L^1$ -data fidelity term instead of the  $L^2$ -data fidelity term in (7.4), i.e., one considers

$$\min_{u \in BV(\Omega)} \|Tu - g\|_{L^1(\Omega)} + \alpha \int_{\Omega} |Du|, \quad (7.5)$$

which we call the  $L^1$ -TV model. In this paper, we are interested in both models, i.e., the  $L^2$ -TV and the  $L^1$ -TV model, and condense them into

$$\min_{u \in BV(\Omega)} \left\{ \mathcal{J}_{\tau}(u; g) := \mathcal{H}_{\tau}(u; g) + \alpha \int_{\Omega} |Du| \right\} \quad (7.6)$$

to obtain a combined model for removing Gaussian or impulsive noise, where  $\mathcal{H}_{\tau}(u; g) := \frac{1}{\tau} \|Tu - g\|_{L^{\tau}(\Omega)}^{\tau}$  for  $\tau = 1, 2$ . Note, that instead of (7.6) one can consider the equivalent

## 7. Automated Parameter Selection for Total Variation Minimization

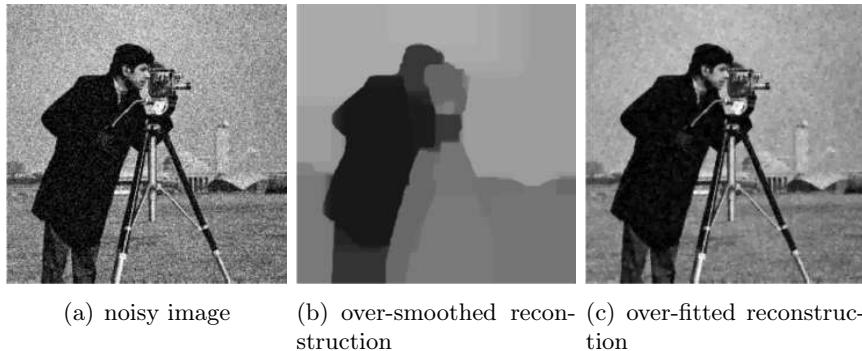
problem

$$\min_{u \in BV(\Omega)} \lambda \mathcal{H}_\tau(u; g) + \int_\Omega |Du|, \quad (7.7)$$

where  $\lambda = \frac{1}{\alpha} > 0$ . Other and different fidelity terms have been considered in connection with other type of noise models, as Poisson noise [64], multiplicative noise [4], Rician noise [43]. For images which are simultaneously contaminated by Gaussian and impulse noise [15] a combined  $L^1$ - $L^2$ -data fidelity term has been recently suggested and demonstrated to work satisfactory [53]. However, in this paper, we concentrate on images degraded by only one type of noise, i.e., either Gaussian noise or one type of impulse noise, and perhaps additionally corrupted by some measurement device.

### 7.1.2. Choice of the Scalar Regularization Parameter

For the reconstruction of such images the proper choice of  $\alpha$  in (7.6) and  $\lambda$  in (7.7) is delicate; cf. Fig. 7.1. In particular, large  $\alpha$  and small  $\lambda$ , which lead to an over-smoothed reconstruction, not only remove noise but also eliminate details in images. On the other hand, small  $\alpha$  and large  $\lambda$  lead to solutions which fit the given data properly but therefore retain noise in homogeneous regions. Hence a good reconstruction can be obtained by choosing  $\alpha$  and respectively  $\lambda$  such that a good compromise of the aforementioned effects are made. There are several ways of how to select  $\alpha$  in (7.6) and equivalently  $\lambda$  in (7.7), such as manually by the trial-and-error method, the unbiased predictive risk estimator method (UPRE) [69, 67], the Stein unbiased risk estimator method (SURE) [82, 38, 10] and its generalizations [34, 40, 45], the generalized cross-validation method (GCV) [48, 66, 67, 78], the L-curve method [49, 50], the discrepancy principle [70], and the variational Bayes' approach [6]. Further parameter selection methods for general inverse problems can be found for example in [41, 42, 88, 89].



**Figure 7.1:** Reconstruction of an image corrupted by Gaussian white noise with a “relatively” large parameter  $\alpha$  in (b) and a “relatively” small parameter  $\alpha$  in (c).

Based on a training set of pairs  $(g_k, \hat{u}_k)$ , for  $k = 1, 2, \dots, N \in \mathbb{N}$ , where  $g_k$  is the noisy observation and  $\hat{u}_k$  represents the original image, for example in [16, 33, 61] bilevel optimization approaches have been presented to compute suitable scalar regularization parameters of the corresponding image model. Since in our setting we do not have a training set given, these approaches are not applicable here.

## II. Parameter Selection Methods for Total Variation Models

Applying the discrepancy principle to estimate  $\alpha$  in (7.6) or  $\lambda$  in (7.7), the image restoration problem can be formulated as a constrained optimization problem of the form

$$\min_{u \in BV(\Omega)} \int_{\Omega} |Du| \quad \text{subject to (s.t.)} \quad \mathcal{H}_{\tau}(u; g) = \mathcal{B}_{\tau} \quad (7.8)$$

where  $\mathcal{B}_{\tau} := \frac{\nu_{\tau}}{\tau} |\Omega|$  with  $\nu_{\tau} > 0$  being here a constant depending on the underlying noise,  $\tau = 1, 2$ , and  $|\Omega|$  denoting the volume of  $\Omega$ ; see Section 7.3 for more details. Note, that here we assume to know a-priori the noise level. In real applications this means that possibly in a first step a noise estimation has to be performed before the discrepancy principle may be used. However, in general it is easier to estimate the noise level than the regularization parameter [18].

The constrained minimization problem (7.8) is naturally linked to the unconstrained minimization problem (7.7) and accordingly to (7.6). In particular, there exists a constant  $\lambda \geq 0$  such that the unconstrained problem (7.7) is equivalent to the constrained problem (7.8) if  $T$  does not annihilate constant functions, i.e.,  $T \in \mathcal{L}(L^2(\Omega))$  is such that  $T \cdot 1 = 1$ ; see Section 7.2 for more details. Several methods based on the discrepancy principle and problem (7.8) with  $\tau = 2$  have been proposed in the literature, see for example [9, 18, 51, 92] and references therein, while not so much attention has been given to the case  $\tau = 1$ , see for example [73, 91].

### 7.1.3. Spatially Adaptive Parameter

Note, that a scalar regularization parameter might not be the best choice for every image restoration problem, since images usually have large homogeneous regions as well as parts with a lot of details. Actually it seems obvious that  $\alpha$  should be small, or  $\lambda$  should be large, in parts with small features in order to preserve the details. On the contrary  $\alpha$  should be large, or  $\lambda$  should be small, in homogeneous parts to remove noise considerable. With such a choice of a spatially varying weight we expect better reconstructions than with a globally constant parameter, as demonstrated for example in [37, 58]. This motivated to consider multiscale total variation models with spatially varying parameters initially suggested in [80]. The multiscale version of (7.6) reads as

$$\min_{u \in BV(\Omega)} \mathcal{H}_{\tau}(u; g) + \int_{\Omega} \alpha(x) |Du| \quad (7.9)$$

while for (7.7) one writes

$$\min_{u \in BV(\Omega)} \frac{1}{\tau} \int_{\Omega} \lambda(x) |Tu - g|^{\tau} dx + \int_{\Omega} |Du|, \quad (7.10)$$

and in the sequel we refer to (7.9) and (7.10) as the multiscale  $L^{\tau}$ -TV model.

In [84] the influence of the scale of an image feature on the choice of  $\alpha$  is studied and the obtained observations were later used in [83] to construct an updating scheme of  $\alpha$ . Based on (7.10) in [8] a piecewise constant function  $\lambda$ , where the pieces are defined by a partitioning of the image due to a pre-segmentation, is determined. In particular, for each segment a scalar  $\lambda_i$ ,  $i = 1, \dots, \#\text{pieces}$  is computed by Uzawa's method [27].

Later it was noticed that stable choices of  $\lambda$  respectively  $\alpha$  should incorporate statistical properties of the noise. In this vein, in [2, 37, 44] for the problem (7.10) automated update rules for  $\lambda$  based on statistics of local constraints were proposed. In [44] a two level approach

## 7. Automated Parameter Selection for Total Variation Minimization

for variational denoising is considered, where in the first level noise and relevant texture are isolated in order to compute local constraints based on local variance estimation. In the second level a gradient descent method and an update formula for  $\lambda(x)$  derived from the Euler-Lagrange equation is utilized. An adaptation of this approach to multiplicative noise can be found in [65]. For convolution type of problems in [2] based on an estimate of the noise variance for each pixel an automatic updating scheme of  $\lambda$  using Uzawa's method is created. This approach is improved in [37] by determining the fidelity weights due to the Gumbel statistic for the maximum of a finite number of random variables associated with localized image residuals and by incorporating hierarchical image decompositions, proposed in [86, 87], to speed up the iterative parameter adjustment process. An adaptation of this approach to a total variation model with  $L^1$  local constraints is studied in [58]. A different approach has been proposed in [85] for image denoising only, where nonlocal means [14] are used to create a nonlocal data fidelity term. While in all these approaches the adjustment of  $\lambda$  relies on the output of  $T$  being a deteriorated image again, in [54] the method of [37] is adjusted to the situation where  $T$  is an orthogonal wavelet transform or Fourier transform. Very recently also bilevel optimisation approaches are considered for computing spatially adaptive weights [26, 56, 57].

### 7.1.4. Contribution

Our first contribution of this paper is to present a method which automatically computes the regularization parameter  $\alpha$  in (7.6) based on (7.8) for  $\tau = 1$  as well as for  $\tau = 2$ . Our approach is motivated by the parameter selection algorithm presented in [18], which was originally introduced for  $L^2$ -TV image denoising only, i.e., when  $T = I$ , where  $I$  denotes the identity operator. In this setting the algorithm in [18] is shown to converge to a parameter  $\alpha^*$  such that the corresponding minimizer  $u_{\alpha^*}$  of (7.6) is also a solution of (7.8). The proof relies on the nonincrease of the function  $\alpha \mapsto \frac{\mathcal{H}_2(u_\alpha; g)}{\mathcal{B}_2}$ . However, this important property does not hold for operators  $T \neq I$  in general. Nevertheless, we generalize the algorithm from [18] to problems of the type (7.6) for  $\tau = 1, 2$  and for general linear bounded operators  $T$ , e.g.,  $T$  might be a convolution type of operator. Utilizing an appropriate update of  $\alpha$ , which is different than the one used in [18], we are able to show analytically and numerically that our approach indeed converges to the desired regularization parameter. Further, besides the general applicability of our proposed method it even possesses advantages for the case  $\tau = 2$  and  $T = I$  over the algorithm from [18] with respect to convergence. More precisely, in our numerics it turned out that our proposed method always needs less or at least the same number of iterations as the algorithm from [18] till termination.

Motivated by multiscale total variation minimization, the second contribution of this paper is concerned with the automated selection of a suitable spatially varying  $\alpha$  for the optimization problem in (7.9) for  $\tau = 1, 2$ . Based on our considerations for an automatic scalar regularization parameter selection, we present algorithms where the adjustment of a locally varying  $\alpha$  is fully automatic. Differently to the scalar case the adjustment of  $\alpha$  is now based on local constraints, similarly as already considered for example in [2, 37, 58]. However, our approach differs significantly from these previous works, where problem (7.10) is considered and Uzawa's method or an Uzawa-like method is utilized for the update of the spatially varying parameter. Note, that in Uzawa's method an additional parameter has to be introduced and chosen accordingly. We propose an update-scheme of  $\alpha$  which does not need any additional parameter and hence is not similar to Uzawa's method. Moreover, differently to the approaches in

## II. Parameter Selection Methods for Total Variation Models

[37, 58] where the initial regularization parameter  $\lambda > 0$  has to be set sufficiently small, in our approach any initial  $\alpha > 0$  is allowed. In this sense is our algorithm even more general than the ones presented in [37, 58].

### 7.1.5. Outline of the Paper

The remaining of the paper is organized as follows: In Section 7.2 we revisit and discuss the connection between the constrained minimization problem (7.8) and the unconstrained optimization problem (7.6). Section 7.3 is devoted to the automated scalar parameter selection. In particular, we present our proposed method and analyze its convergence behavior. Based on local constraints we describe in Section 7.4 our new locally adapted total variation algorithm in detail. Algorithms for performing total variation minimization for spatially varying  $\alpha$  are presented in Section 7.5 where also their convergence properties are studied. To demonstrate the performance of the new algorithms we present in Section 7.6 numerical experiments for image denoising and image deblurring. Finally, in Section 7.7 conclusions are drawn.

## 7.2. Constrained versus Unconstrained Minimization Problem

In this section we discuss the connection between the unconstrained minimization problem (7.6) and the constrained optimization problem (7.8). For this purpose we introduce the following basic terminology. Let  $\mathcal{V}$  be a locally convex space,  $\mathcal{V}'$  its topological dual, and  $\langle \cdot, \cdot \rangle$  the bilinear canonical pairing over  $\mathcal{V} \times \mathcal{V}'$ . The domain of a functional  $\mathcal{J} : \mathcal{V} \rightarrow \mathbb{R} \cup \{+\infty\}$  is defined as the set

$$\text{Dom}(\mathcal{J}) := \{v \in \mathcal{V} : \mathcal{J}(v) < \infty\}.$$

A functional  $\mathcal{J}$  is called *lower semicontinuous* (l.s.c) if for every weakly convergent subsequence  $v^{(n)} \rightharpoonup \hat{v}$  we have

$$\liminf_{v^{(n)} \rightharpoonup \hat{v}} \mathcal{J}(v^{(n)}) \geq \mathcal{J}(\hat{v}).$$

For a convex functional  $\mathcal{J} : \mathcal{V} \rightarrow \mathbb{R} \cup \{+\infty\}$ , we define the *subdifferential* of  $\mathcal{J}$  at  $v \in \mathcal{V}$ , as the set valued function  $\partial\mathcal{J}(v) = \emptyset$  if  $\mathcal{J}(v) = \infty$ , and otherwise as

$$\partial\mathcal{J}(v) = \{v^* \in \mathcal{V}' : \langle v^*, u - v \rangle + \mathcal{J}(v) \leq \mathcal{J}(u) \quad \forall u \in \mathcal{V}\}.$$

For any operator  $T$  we denote by  $T^*$  its adjoint and by  $\mathcal{L}(L^2(\Omega))$  we denote the space of linear and continuous operators from  $L^2(\Omega)$  to  $L^2(\Omega)$ . Moreover,  $g_\Omega$  describes the average value of the function  $g \in L^1(\Omega)$  in  $\Omega$  defined by  $g_\Omega := \frac{1}{|\Omega|} \int_\Omega g(x) \, dx$ .

**Theorem 7.1.** *Assume that  $T \in \mathcal{L}(L^2(\Omega))$  does not annihilate constant functions, i.e.,  $T1_\Omega \neq 0$ , where  $1_\Omega(x) = 1$  for  $x \in \Omega$ . Then the problem*

$$\min_{u \in BV(\Omega)} \int_\Omega |Du| \quad \text{s.t.} \quad \mathcal{H}_\tau(u; g) \leq \mathcal{B}_\tau \tag{7.11}$$

has a solution for  $\tau = 1, 2$ .

*Proof.* For a proof we refer the reader to [20] and [58]. □

Moreover, we have the following statement.

## 7. Automated Parameter Selection for Total Variation Minimization

**Proposition 7.2.** Assume that  $T \in \mathcal{L}(L^2(\Omega))$  is such that  $T \cdot 1 = 1$  and  $\nu_\tau |\Omega| \leq \|g - g_\Omega\|_{L^\tau(\Omega)}^\tau$ . Then problem (7.11) is equivalent to the constrained minimization problem (7.8) for  $\tau = 1, 2$ .

*Proof.* For  $\tau = 2$  the statement is shown in [20]. We state the proof for  $\tau = 1$  by noting it follows similar arguments as for  $\tau = 2$ . Let  $\tilde{u}$  be a solution of (7.11). Note, that there exists  $u \in BV(\Omega)$  such that  $\tilde{u} = u + g_\Omega$ . We consider now the continuous function  $f(s) = \|T(su + g_\Omega) - g\|_{L^1(\Omega)}$  for  $s \in [0, 1]$ . Note that  $f(1) = \|T\tilde{u} - g\|_{L^1(\Omega)} \leq \nu_1 |\Omega|$  and  $f(0) = \|Tg_\Omega - g\|_{L^1(\Omega)} = \|g - g_\Omega\|_{L^1(\Omega)} \geq \nu_1 |\Omega|$ , since  $T \cdot 1 = 1$ , and hence there exists some  $s \in [0, 1]$  such that  $f(s) = \nu_1 |\Omega|$ . Set  $u' = su$  which satisfies  $\|Tu' - g\|_{L^1(\Omega)} = \nu_1 |\Omega|$  and

$$\int_{\Omega} |Du'| = s \int_{\Omega} |Du| \leq \liminf_{n \rightarrow \infty} \int_{\Omega} |Du_n|,$$

where  $(u_n)_n$  is a minimizing sequence of (7.8). Hence  $u'$  is a solution of (7.8).  $\square$

Now we are able to argue the equivalence of the problems (7.6) and (7.8).

**Theorem 7.3.** Let  $T \in \mathcal{L}(L^2(\Omega))$  be such that  $T \cdot 1 = 1$  and  $\nu_\tau |\Omega| \leq \|g - g_\Omega\|_{L^\tau(\Omega)}^\tau$ . Then there exists  $\alpha \geq 0$  such that the constrained minimization problem (7.8) is equivalent to the unconstrained problem (7.6), i.e.,  $u$  is a solution of (7.8) if and only if  $u$  solves (7.6).

*Proof.* For  $\tau = 2$  the proof can be found in [20, Prop. 2.1]. By similar arguments one can show the statement for  $\tau = 1$ , which we state here.

Set  $\mathcal{R}(u) := \int_{\Omega} |Du|$  and

$$\mathcal{G}(u) = \begin{cases} +\infty & \text{if } \|u - g\|_{L^1(\Omega)} > \nu_1 |\Omega|, \\ 0 & \text{if } \|u - g\|_{L^1(\Omega)} \leq \nu_1 |\Omega|. \end{cases}$$

Notice, that  $\mathcal{R}$  and  $\mathcal{G}$  are convex l.s.c functions and problem (7.11) is equivalent to  $\min_u \mathcal{R}(u) + \mathcal{G}(Tu)$ . We have  $\text{Dom}(\mathcal{R}) = BV(\Omega) \cap L^2(\Omega)$  and  $\text{Dom}(\mathcal{G}) = \{u \in L^2(\Omega) : \mathcal{G}(u) < +\infty\}$ . Since  $g \in \overline{T \text{Dom}(\mathcal{R})}$ , there exists  $\tilde{u} \in \text{Dom}(\mathcal{R})$  with  $\|T\tilde{u} - g\|_{L^1(\Omega)} \leq \nu_1 |\Omega|/2$ . As  $T \in \mathcal{L}(L^2(\Omega))$  is continuous,  $\mathcal{G} \circ T$  is continuous at  $\tilde{u}$ . Hence, by [39, Prop. 5.6, p. 26] we obtain

$$\partial(\mathcal{R} + \mathcal{G} \circ T)(u) = \partial\mathcal{R}(u) + \partial(\mathcal{G} \circ T)(u)$$

for all  $u$ . Further,  $\mathcal{G}$  is continuous at  $T\tilde{u}$ , and hence by [39, Prop. 5.7, p. 27] we have for all  $u$ ,

$$\partial(\mathcal{G} \circ T)(u) = T^* \partial\mathcal{G}(Tu)$$

where  $\partial\mathcal{G}(u) = \{0\}$  if  $\|u - g\|_{L^1(\Omega)} < \nu_1 |\Omega|$  and  $\partial\mathcal{G}(u) = \{\alpha \partial(\|u - g\|_{L^1(\Omega)}) : \alpha \geq 0\}$  if  $\|u - g\|_{L^1(\Omega)} = \nu_1 |\Omega|$ .

If  $u$  is a solution of (7.11) and hence of (7.8), then

$$0 \in \partial(\mathcal{R} + \mathcal{G} \circ T)(u) = \partial\mathcal{R}(u) + T^* \partial\mathcal{G}(Tu).$$

Since any solution of (7.8) satisfies  $\|Tu - g\|_{L^1(\Omega)} = \nu_1 |\Omega|$ , this shows that there exists an  $\alpha \geq 0$  such that

$$0 \in \partial\mathcal{R}(u) + \alpha T^* \partial(\|Tu - g\|_{L^1(\Omega)}).$$

Hence for this  $\alpha \geq 0$ ,  $u$  is a minimizer of the problem in (7.6).

Conversely, a minimizer  $u$  of (7.6) with the above  $\alpha$  is obviously a solution of (7.8) with  $\|Tu - g\|_{L^1(\Omega)} = \nu_1 |\Omega|$ . This concludes the proof.  $\square$

## II. Parameter Selection Methods for Total Variation Models

Note, that  $\|Tu_\alpha - g\|_{L^1(\Omega)}$  is (only) convex with respect to  $Tu_\alpha$ , and hence a minimizer of (7.5) is in general not unique even in the simple case when  $T = I$ , i.e., for two minimizers  $u_\alpha^1$  and  $u_\alpha^2$  in general we have  $Tu_\alpha^1 \neq Tu_\alpha^2$ . On the contrary,  $\|Tu_\alpha - g\|_{L^2(\Omega)}^2$  is strictly convex with respect to  $Tu_\alpha$ , i.e., for two minimizers  $u_\alpha^1$  and  $u_\alpha^2$  of (7.4) we have  $Tu_\alpha^1 = Tu_\alpha^2$ . Moreover, the function  $\alpha \mapsto \mathcal{H}_1(u_\alpha; g)$  is in general not continuous [23], while  $\alpha \mapsto \mathcal{H}_2(u_\alpha; g)$  indeed is continuous [20], where  $u_\alpha$  is a respective minimizer of (7.6). Hence we have the following further properties:

**Lemma 7.4.** *Let  $u_\alpha$  be a minimizer of (7.6) then  $\alpha \mapsto \mathcal{H}_\tau(u_\alpha; g)$  is nondecreasing for  $\tau = 1, 2$ . Moreover,  $\alpha \mapsto \mathcal{H}_2(u_\alpha; g)$  maps  $\mathbb{R}^+$  onto  $[0, \|g - g_\Omega\|_{L^2(\Omega)}^2]$ .*

*Proof.* For a proof see [20, 23].  $\square$

**Proposition 7.5.** *If  $u_{\alpha_i}$  is a minimizer of*

$$\mathcal{E}(u, \alpha_i) := \|u - g\|_{L^2(\Omega)}^2 + \alpha_i \int_\Omega |Du|$$

for  $i = 1, 2$ , then we have

$$\|u_{\alpha_1} - u_{\alpha_2}\|_{L^2(\Omega)} \leq C \|g - g_\Omega\|_{L^2(\Omega)}.$$

$$\text{with } C := \min \left\{ 2 \left| \frac{\alpha_2 - \alpha_1}{\alpha_2 + \alpha_1} \right|, \left| \frac{\alpha_2 - \alpha_1}{\alpha_2 + \alpha_1} \right|^{\frac{1}{2}} \right\}$$

*Proof.* By [7, Lemma 10.2] we have

$$\begin{aligned} \frac{1}{\alpha_1} \|u_{\alpha_1} - u_{\alpha_2}\|_{L^2(\Omega)}^2 &\leq \frac{1}{\alpha_1} (\mathcal{E}(u_{\alpha_2}, \alpha_1) - \mathcal{E}(u_{\alpha_1}, \alpha_1)) \\ \frac{1}{\alpha_2} \|u_{\alpha_1} - u_{\alpha_2}\|_{L^2(\Omega)}^2 &\leq \frac{1}{\alpha_2} (\mathcal{E}(u_{\alpha_1}, \alpha_2) - \mathcal{E}(u_{\alpha_2}, \alpha_2)). \end{aligned}$$

Summing up these two inequalities yields

$$\left( \frac{1}{\alpha_1} + \frac{1}{\alpha_2} \right) \|u_{\alpha_1} - u_{\alpha_2}\|_{L^2(\Omega)}^2 \leq \left( \frac{1}{\alpha_1} - \frac{1}{\alpha_2} \right) \left( \|u_{\alpha_2} - g\|_{L^2(\Omega)}^2 - \|u_{\alpha_1} - g\|_{L^2(\Omega)}^2 \right)$$

which implies

$$\|u_{\alpha_1} - u_{\alpha_2}\|_{L^2(\Omega)}^2 \leq \frac{\alpha_2 - \alpha_1}{\alpha_1 + \alpha_2} \left( \|u_{\alpha_2} - g\|_{L^2(\Omega)}^2 - \|u_{\alpha_1} - g\|_{L^2(\Omega)}^2 \right). \quad (7.12)$$

By the nondecrease and boundedness of the function  $\alpha \mapsto \mathcal{H}_2(u_\alpha; g)$ , see Lemma 7.4, it follows

$$\|u_{\alpha_1} - u_{\alpha_2}\|_{L^2(\Omega)}^2 \leq \left| \frac{\alpha_2 - \alpha_1}{\alpha_1 + \alpha_2} \right| \|g - g_\Omega\|_{L^2(\Omega)}^2. \quad (7.13)$$

On the other hand inequality (7.12) implies

$$\|u_{\alpha_1} - u_{\alpha_2}\|_{L^2(\Omega)} \leq \left| \frac{\alpha_2 - \alpha_1}{\alpha_1 + \alpha_2} \right| (\|u_{\alpha_2} - g\|_{L^2(\Omega)} + \|u_{\alpha_1} - g\|_{L^2(\Omega)}),$$

where we used the binomial formula  $a^2 - b^2 = (a + b)(a - b)$  for  $a, b \in \mathbb{R}$  and the triangle inequality. Using Lemma 7.4 yields

$$\|u_{\alpha_1} - u_{\alpha_2}\|_{L^2(\Omega)} \leq 2 \left| \frac{\alpha_2 - \alpha_1}{\alpha_1 + \alpha_2} \right| \|g - g_\Omega\|_{L^2(\Omega)}. \quad (7.14)$$

The assertion follows then from (7.13) and (7.14).  $\square$

## 7. Automated Parameter Selection for Total Variation Minimization

**Remark 7.6.** Without loss of generality let  $\alpha_2 \geq \alpha_1$  in Proposition 7.5, then we easily check that

$$C = \begin{cases} \left(\frac{\alpha_2 - \alpha_1}{\alpha_2 + \alpha_1}\right)^{\frac{1}{2}} & \text{if } \alpha_2 > \frac{5}{3}\alpha_1, \\ \frac{1}{2} & \text{if } \alpha_2 = \frac{5}{3}\alpha_1, \\ 2^{\frac{\alpha_2 - \alpha_1}{\alpha_2 + \alpha_1}} & \text{otherwise.} \end{cases}$$

### 7.3. Automated Scalar Parameter Selection

In order to find a suitable regularization parameter  $\alpha > 0$  of the minimization problem (7.6) we consider the corresponding constrained optimization problem (7.8). Throughout the paper we assume that  $T$  does not annihilate constant function, which guarantees the existence of a minimizer of the considered optimization problems; see Section 7.2. We recall, that in the constraint of (7.8) the value  $\mathcal{B}_\tau$  is defined as  $\mathcal{B}_\tau = \frac{\nu_\tau}{\tau} |\Omega|$ , where  $\nu_\tau \in \mathbb{R}$  is a statistical value depending on the underlying noise and possibly on the original image.

#### 7.3.1. Statistical Characterization of the Noise

Let us characterize the noise corrupting the image in more details by making similar considerations as in [58, Section 2]. Note, that at any point  $x \in \Omega$  the contaminated image  $g(x) = \mathcal{N}(T\hat{u})(x)$  is a stochastic observation, which depends on the underlying noise. Two important measures to characterize noise are the expected absolute value and the variance, which we denote by  $\nu_1$  and  $\nu_2$  respectively. For images contaminated by Gaussian white noise with standard deviation  $\sigma$ , we typically set  $\tau = 2$  and  $\nu_2 = \sigma^2$ . If the image is instead corrupted by impulse noise, then we set  $\tau = 1$  and we have to choose  $\nu_1$  properly. In particular, for salt-and-pepper noise  $\nu_1 \in [\min\{r_1, r_2\}, \max\{r_1, r_2\}]$ , while for random-valued impulse noise  $\nu_1$  should be a value in the interval  $[\frac{r}{4}, \frac{r}{2}]$ , where we used that for any point  $x \in \Omega$  we have  $T\hat{u}(x) \in [0, 1]$ ; cf. [58]. Here  $\nu_1$  seems to be fixed, while actually  $\nu_1$  depends on the true (unknown) image  $\hat{u}$ . In particular, for salt-and-pepper noise the expected absolute value is given by

$$\nu_1(\hat{u}) := r_2 - (r_2 - r_1) \frac{1}{|\Omega|} \int_{\Omega} (T\hat{u})(x) \, dx \quad (7.15)$$

and for random-valued impulse noise we have

$$\nu_1(\hat{u}) := \frac{1}{|\Omega|} \int_{\Omega} r \left( (T\hat{u})(x)^2 - (T\hat{u})(x) + \frac{1}{2} \right) \, dx. \quad (7.16)$$

However, instead of considering the constraint  $\mathcal{H}_\tau(u; g) = \mathcal{B}_\tau(u)$  in (7.8), which results in a quite nonlinear problem, in our numerics we choose a reference image and compute an approximate value  $\mathcal{B}_\tau$ . Since our proposed algorithms are of iterative nature (see APS- and pAPS-algorithm below), it makes sense to choose the current approximation as the reference image, i.e., the reference image changes during the iterations. Note, that for salt-and-pepper noise with  $r_1 = r_2$  the expected absolute value becomes independent of  $\hat{u}$  and hence  $\nu_1 = r_1$ . In case of Gaussian noise  $\nu_\tau$  and  $\mathcal{B}_\tau$  are independent of  $\hat{u}$  too. Nevertheless, in order to keep the paper concise, in the sequel instead of  $\nu_\tau$  and  $\mathcal{B}_\tau$  we often write  $\nu_\tau(\tilde{u})$  and  $\mathcal{B}_\tau(\tilde{u})$ , where  $\tilde{u}$  represents a reference image approximating  $\hat{u}$ , even if the values may actually be independent from the image.

## II. Parameter Selection Methods for Total Variation Models

### 7.3.2. Automated Parameter Selection Strategy

In order to determine a suitable regularization parameter  $\alpha$  in [18] an algorithm for solving the constrained minimization problem (7.8) for  $T = I$  and  $\tau = 2$  is proposed, i.e., in the presence of Gaussian noise with zero mean and standard deviation  $\sigma$ . This algorithm relies on the fact that  $\alpha \mapsto \mathcal{H}_2(u_\alpha; g)$  is nondecreasing, which leads to the following iterative procedure.

**Chambolle's parameter selection (CPS):** Choose  $\alpha_0 > 0$  and set  $n := 0$ .

1) Compute

$$u_{\alpha_n} \in \arg \min_{u \in BV(\Omega)} \|u - g\|_{L^2(\Omega)}^2 + 2\alpha_n \int_{\Omega} |Du|$$

2) Update  $\alpha_{n+1} := \frac{\sigma\sqrt{|\Omega|}}{\|u_{\alpha_n} - g\|_{L^2(\Omega)}} \alpha_n$ .

3) Stop or set  $n := n + 1$  and return to step 1).

For the minimization of the optimization problem in step 1) in [18] a method based on the dual formulation of the total variation is used. However, we note that any other algorithm for total variation minimization might be used for solving this minimization problem. The CPS-algorithm generates a sequence  $(u_{\alpha_n})_n$  such that for  $n \rightarrow \infty$ ,  $\|u_{\alpha_n} - g\|_{L^2(\Omega)} \rightarrow \sigma\sqrt{|\Omega|}$  and  $u_{\alpha_n}$  converges to the unique solution of (7.8) with  $T = I$  and  $\tau = 2$  [18]. The proof relies on the fact that the function  $\alpha \rightarrow \frac{\|u_\alpha - g\|_{L^2(\Omega)}}{\alpha}$  is nonincreasing. Note, that this property does not hold in general for operators  $T \neq I$ .

#### 7.3.2.1. The p-Adaptive Algorithm

We generalize now the CPS-algorithm to optimization problems of the type (7.8) for  $\tau = 1, 2$  and for general operators  $T$ . In order to keep or obtain appropriate convergence properties, we need the following two conditions to be satisfied. Firstly, the function  $\alpha \mapsto \mathcal{H}_\tau(u_\alpha; g)$  has to be monotonic, which is the case due to Lemma 7.4. Secondly, in each iteration  $n$  the parameter  $\alpha_n$  has to be updated such that  $(\mathcal{H}_\tau(u_{\alpha_n}; g))_n$  is monotonic and bounded by  $(\mathcal{B}_\tau(u_{\alpha_n}))_n$ . More precisely, if  $\mathcal{H}_\tau(u_{\alpha_0}; g) \leq \mathcal{B}_\tau(u_{\alpha_0})$  then there has to exist an  $\alpha_{n+1} \geq \alpha_n$  such that  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{B}_\tau(u_{\alpha_{n+1}})$ , while if  $\mathcal{H}_\tau(u_{\alpha_0}; g) > \mathcal{B}_\tau(u_{\alpha_0})$  then there has to exist an  $\alpha_{n+1} \leq \alpha_n$  such that  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \geq \mathcal{B}_\tau(u_{\alpha_{n+1}})$ . This holds true by setting in every iteration

$$\alpha_{n+1} := \left( \frac{\mathcal{B}_\tau(u_{\alpha_n})}{\mathcal{H}_\tau(u_{\alpha_n}; g)} \right)^p \alpha_n \quad (7.17)$$

together with an appropriate choice of  $p \geq 0$ . In particular, there exists always a  $p \geq 0$  such that this condition is satisfied.

**Proposition 7.7.** Assume  $\|g - g_\Omega\|_{L^\tau(\Omega)}^\tau \geq \nu_\tau |\Omega|$  and  $\alpha_{n+1}$  is defined as in (7.17).

- (i) If  $\alpha_n > 0$  such that  $\mathcal{H}_\tau(u_{\alpha_n}; g) = \mathcal{B}_\tau(u_{\alpha_n})$ , then for all  $p \in \mathbb{R}$  we have that  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{B}_\tau(u_{\alpha_{n+1}})$ .
- (ii) If  $\alpha_n > 0$  such that  $0 < \mathcal{H}_\tau(u_{\alpha_n}; g) < \mathcal{B}_\tau(u_{\alpha_n})$ , then there exist  $p \geq 0$  with  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{B}_\tau(u_{\alpha_{n+1}})$ .

## 7. Automated Parameter Selection for Total Variation Minimization

(iii) If  $\alpha_n > 0$  such that  $\mathcal{H}_\tau(u_{\alpha_n}; g) > \mathcal{B}_\tau(u_{\alpha_n})$ , then there exist  $p \geq 0$  with  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \geq \mathcal{B}_\tau(u_{\alpha_{n+1}})$ .

*Proof.* The assertion immediately follows by noting that for  $p = 0$  we have  $\alpha_{n+1} = \alpha_n$ .  $\square$

Taking these considerations into account, a generalization of the CPS-algorithm can be formulated as the following  $p$ -adaptive automated parameter selection algorithm:

**pAPS-algorithm:** Choose  $\alpha_0 > 0$ ,  $p := p_0 > 0$ , and set  $n := 0$ .

- 1) Compute  $u_{\alpha_n} \in \arg \min_{u \in BV(\Omega)} \mathcal{J}_\tau(u; g)$
- 2) Update  $\alpha_{n+1} := \left( \frac{\mathcal{B}_\tau(u_{\alpha_n})}{\mathcal{H}_\tau(u_{\alpha_n}; g)} \right)^p \alpha_n$  if  $\mathcal{H}_\tau(u_{\alpha_n}; g) > 0$  and continue with step 3). Otherwise increase  $\alpha_n$ , e.g.,  $\alpha_n := 10\alpha_n$ , and go to step 1).
- 3) Compute  $u_{\alpha_{n+1}} \in \arg \min_{u \in BV(\Omega)} \mathcal{J}_\tau(u; g)$
- 4) a) if  $\mathcal{H}_\tau(u_{\alpha_0}; g) \leq \mathcal{B}_\tau(u_{\alpha_0})$ 
  - (i) if  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{B}_\tau(u_{\alpha_{n+1}})$  go to step 5)
  - (ii) if  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) > \mathcal{B}_\tau(u_{\alpha_{n+1}})$ , decrease  $p$ , e.g., set  $p := p/2$ , and go to step 2)
- b) if  $\mathcal{H}_\tau(u_{\alpha_0}; g) > \mathcal{B}_\tau(u_{\alpha_0})$ 
  - (i) if  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \geq \mathcal{B}_\tau(u_{\alpha_{n+1}})$  go to step 5)
  - (ii) if  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) < \mathcal{B}_\tau(u_{\alpha_{n+1}})$ , decrease  $p$ , e.g., set  $p := p/2$ , and go to step 2)
- 5) Stop or set  $n := n + 1$  and return to step 2).

Note, that due to the dependency of  $\alpha_{n+1}$  on  $p$  a proper  $p$  cannot be explicitly computed, but only iteratively, as in the pAPS-algorithm.

The initial  $p_0 > 0$  can be chosen arbitrarily. However, we suggest to choose it sufficiently large in order to keep the number of iterations small. In particular in our numerical experiments in Section 7.6 we set  $p_0 = 32$ , which seems large enough to us.

**Proposition 7.8.** *The pAPS-algorithm generates monotone sequences  $(\alpha_n)_n$  and  $(\mathcal{H}_\tau(u_{\alpha_n}; g))_n$  such that*

$(\mathcal{H}_\tau(u_{\alpha_n}; g))_n$  is bounded. Moreover, if  $\mathcal{H}_\tau(u_{\alpha_0}; g) > \mathcal{B}_\tau(u_{\alpha_0})$  or  $\mathcal{B}_\tau(u_\alpha) \leq \frac{1}{\tau} \|g - g_\Omega\|_\tau^\tau$  for all  $\alpha > 0$ , then  $(\alpha_n)_n$  is also bounded.

*Proof.* If  $\mathcal{H}_\tau(u_{\alpha_0}; g) > \mathcal{B}_\tau(u_{\alpha_0})$ , then by induction and Lemma 7.4 one shows that  $0 < \alpha_{n+1} \leq \alpha_n$  and  $0 \leq \mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{H}_\tau(u_{\alpha_n}; g)$  for all  $n \in \mathbb{N}$ . Consequently  $(\alpha_n)_n$  and  $(\mathcal{H}_\tau(u_{\alpha_n}; g))_n$  are monotonically decreasing and bounded.

If  $\mathcal{H}_\tau(u_{\alpha_0}; g) \leq \mathcal{B}_\tau(u_{\alpha_0})$ , due to Lemma 7.4 we have that  $0 < \alpha_n \leq \alpha_{n+1}$  and  $\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{H}_\tau(u_{\alpha_{n+1}}; g)$  for all  $n \in \mathbb{N}$  and hence  $(\alpha_n)_n$  and  $(\mathcal{H}_\tau(u_{\alpha_n}; g))_n$  are monotonically increasing. Since there exists  $\mathcal{B}_\tau^* > 0$  such that  $\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{B}_\tau(u_{\alpha_n}) \leq \mathcal{B}_\tau^*$  for all  $n \in \mathbb{N}$ , see Section 7.3.1,  $(\mathcal{H}_\tau(u_{\alpha_n}; g))_n$  is also bounded. If we additionally assume that  $\mathcal{B}_\tau(u_\alpha) \leq \frac{1}{\tau} \|g - g_\Omega\|_\tau^\tau$  for all  $\alpha > 0$  and we set  $\mathcal{B}_\tau^* := \max_\alpha \mathcal{B}_\tau(u_\alpha)$ , then Theorem 7.3 ensures the existence of an  $\alpha^* \geq 0$  such that  $\mathcal{H}_\tau(u_{\alpha^*}; g) = \mathcal{B}_\tau^*$ . By Lemma 7.4 it follows that  $\alpha_n \leq \alpha^*$  for all  $n \in \mathbb{N}$ , since  $\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{B}_\tau(u_{\alpha_n}) \leq \mathcal{B}_\tau^*$ . Hence,  $(\alpha_n)_n$  is bounded, which finishes the proof.  $\square$

Since any monotone and bounded sequence converges to a finite limit, also  $(\alpha_n)_n$  converges to a finite value if one of the assumptions in Proposition 7.8 holds. For constant  $\mathcal{B}_\tau$  we are

## II. Parameter Selection Methods for Total Variation Models

even able to argue the convergence of the pAPS-algorithm to a solution of the constrained minimization problem (7.8).

**Theorem 7.9.** *Assume that  $\mathcal{B}_\tau(u) \equiv \mathcal{B}_\tau$  is a constant independent of  $u$  and  $\|g - g_\Omega\|_\tau^\tau \geq \nu_\tau |\Omega|$ . Then the pAPS-algorithm generates a sequence  $(\alpha_n)_n$  such that  $\lim_{n \rightarrow \infty} \alpha_n = \bar{\alpha} > 0$  with  $\mathcal{H}_\tau(u_{\bar{\alpha}}; g) = \lim_{n \rightarrow \infty} \mathcal{H}_\tau(u_{\alpha_n}; g) = \mathcal{B}_\tau$  and  $u_{\alpha_n} \rightarrow u_{\bar{\alpha}} \in \arg \min_{u \in X} \mathcal{J}_\tau(u, \bar{\alpha})$  for  $n \rightarrow \infty$ .*

*Proof.* Let us start with assuming that  $\mathcal{H}_\tau(u_{\alpha_0}; g) \leq \mathcal{B}_\tau$ . By induction, we show that  $\alpha_n \leq \alpha_{n+1}$  and  $\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{B}_\tau$ . In particular, if  $\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{B}_\tau$  then  $\alpha_{n+1} = \left(\frac{\mathcal{B}_\tau}{\mathcal{H}_\tau(u_{\alpha_n}; g)}\right)^p \alpha_n > \alpha_n$ , where  $p > 0$  such that  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{B}_\tau$ ; cf. pAPS-algorithm. Then by Lemma 7.4 it follows that

$$\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{B}_\tau.$$

Note, that there exists an  $\alpha^* > 0$  with  $\mathcal{H}_\tau(u_{\alpha^*}; g) = \mathcal{B}_\tau$ , see Theorem 7.3, such that for any  $\alpha \geq \alpha^*$ ,  $\mathcal{H}_\tau(u_\alpha; g) \geq \mathcal{B}_\tau$ ; cf. Lemma 7.4. If  $\alpha_n \geq \alpha^*$ , then  $\mathcal{H}_\tau(u_{\alpha_n}; g) \geq \mathcal{B}_\tau$ . Hence  $\mathcal{H}_\tau(u_{\alpha_n}; g) = \mathcal{B}_\tau$  and  $\alpha_{n+1} = \alpha_n$ . Thus we deduce that the sequences  $(\mathcal{H}_\tau(u_{\alpha_n}; g))_n$  and  $(\alpha_n)_n$  are nondecreasing and bounded. Consequently, there exists an  $\bar{\alpha}$  such that  $\lim_{n \rightarrow \infty} \alpha_n = \bar{\alpha}$  with  $\mathcal{H}_\tau(u_{\bar{\alpha}}; g) = \mathcal{B}_\tau$ . Let  $\tilde{\mathcal{H}} = \lim_{n \rightarrow \infty} \mathcal{H}_\tau(u_{\alpha_n}; g)$ , then  $\tilde{\mathcal{H}} = \mathcal{H}_\tau(u_{\bar{\alpha}}; g) = \mathcal{B}_\tau$ . By the optimality of  $u_{\alpha_n}$  we have that  $0 \in \partial \mathcal{J}_\tau(u_{\alpha_n}, \alpha_n) = \partial \mathcal{H}_\tau(u_{\alpha_n}; g) + \alpha_n \partial \int_\Omega |Du_{\alpha_n}|$ ; see [39, Prop. 5.6 + Eq. (5.21), p.26]. Consequently there exist  $v_{\alpha_n} \in \partial \int_\Omega |Du_{\alpha_n}|$  such that  $-\alpha_n v_{\alpha_n} \in \partial \mathcal{H}_\tau(u_{\alpha_n}; g)$  with  $\lim_{n \rightarrow \infty} v_{\alpha_n} = v_{\bar{\alpha}}$ . By [79, Thm. 24.4, p. 233] we obtain that  $-\bar{\alpha} v_{\bar{\alpha}} \in \partial \mathcal{H}_\tau(u_{\bar{\alpha}}; g)$  with  $v_{\bar{\alpha}} \in \partial \int_\Omega |Du_{\bar{\alpha}}|$  and hence  $0 \in \partial \mathcal{J}_\tau(u_{\bar{\alpha}}, \bar{\alpha})$  for  $n \rightarrow \infty$ .

If  $\mathcal{H}_\tau(u_{\alpha_0}; g) > \mathcal{B}_\tau$ , then as above we can show by induction that  $\alpha_n \geq \alpha_{n+1}$  and  $\mathcal{H}_\tau(u_{\alpha_n}; g) \geq \mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \geq \mathcal{B}_\tau$ . Thus we deduce that  $(\mathcal{H}_\tau(u_{\alpha_n}; g))_n$  and  $(\alpha_n)_n$  are nonincreasing and bounded. Note, that there exists an  $\alpha^* > 0$  with  $\mathcal{H}_\tau(u_{\alpha^*}; g) = \mathcal{B}_\tau$  such that for any  $\alpha \leq \alpha^*$ ,  $\mathcal{H}_\tau(u_\alpha; g) \leq \mathcal{B}_\tau$ . Hence if  $\alpha_n \leq \alpha^*$ , then  $\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{B}_\tau$ . This implies, that  $\mathcal{H}_\tau(u_{\alpha_n}; g) = \mathcal{B}_\tau$  and  $\alpha_{n+1} = \alpha_n$ . The rest of the proof is identical to above.

□

**Remark 7.10.** *The adaptive choice of the value  $p$  in the pAPS-algorithm is fundamental for proving convergence in Theorem 7.9. In particular, the value  $p$  is chosen in dependency of  $\alpha$ , i.e., actually  $p = p(\alpha)$ , such that in the case of a constant  $\mathcal{B}_\tau$  the function  $\alpha \mapsto \frac{\mathcal{H}_\tau(u_\alpha; g)^{p(\alpha)}}{\alpha}$  is nonincreasing; cf. Fig. 7.5(a).*

### 7.3.2.2. The Non p-Adaptive Case

A special case of the pAPS-algorithm accrues when the value  $p$  is not adapted in each iteration but set fixed. For the case  $p = 1$  (fixed) we obtain the following automated parameter selection algorithm.

**APS-algorithm:** Choose  $\alpha_0 > 0$  and set  $n := 0$ .

- 1) Compute  $u_{\alpha_n} \in \arg \min_{u \in BV(\Omega)} \mathcal{J}_\tau(u, \alpha_n)$
- 2) Update  $\alpha_{n+1} := \frac{\mathcal{B}_\tau(u_{\alpha_n})}{\mathcal{H}_\tau(u_{\alpha_n}; g)} \alpha_n$  if  $\mathcal{H}_\tau(u_{\alpha_n}; g) > 0$  and continue with step 3). Otherwise increase  $\alpha_n$ , e.g.,  $\alpha_n := 10\alpha_n$ , and go to step 1).
- 3) Stop or set  $n := n + 1$  and return to step 1).

## 7. Automated Parameter Selection for Total Variation Minimization

Even in this case, although under certain assumptions, we can immediately argue the convergence of this algorithm.

**Theorem 7.11.** *For  $\alpha > 0$  let  $u_\alpha$  be a minimizer of  $\mathcal{J}_\tau(u, \alpha)$ . Assume that  $\mathcal{B}_\tau(u) \equiv \mathcal{B}_\tau$  is a constant independent of  $u$ , the function  $\alpha \mapsto \frac{\mathcal{H}_\tau(u_\alpha; g)}{\alpha}$  is nonincreasing, and  $\|g - g_\omega\|_\tau^\tau \geq \nu_\tau |\Omega|$ . Then the APS-algorithm generates a sequence  $(\alpha_n)_n \subset \mathbb{R}^+$  such that  $\lim_{n \rightarrow \infty} \alpha_n = \bar{\alpha} > 0$ ,  $\lim_{n \rightarrow \infty} \mathcal{H}_\tau(u_{\alpha_n}; g) = \mathcal{B}_\tau$  and  $u_{\alpha_n}$  converges to  $u_{\bar{\alpha}} \in \arg \min_{u \in BV(\Omega)} \mathcal{J}(u, \bar{\alpha})$  for  $n \rightarrow \infty$ .*

*Proof.* We only consider the case when  $\mathcal{H}_\tau(u_{\alpha_0}; g) \leq \mathcal{B}_\tau$  by noting that the case  $\mathcal{H}_\tau(u_{\alpha_0}; g) > \mathcal{B}_\tau$  can be shown analogous. By induction, we can show that  $\alpha_n \leq \alpha_{n+1}$  and  $\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{B}_\tau$ . More precisely, if  $\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{B}_\tau$  then  $\alpha_{n+1} = \frac{\mathcal{B}_\tau}{\mathcal{H}_\tau(u_{\alpha_n}; g)} \alpha_n \geq \alpha_n$  and by Lemma 7.4 it follows that  $\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{H}_\tau(u_{\alpha_{n+1}}; g)$ . Moreover, by the assumption that  $\alpha \mapsto \frac{\mathcal{H}_\tau(u_\alpha; g)}{\alpha}$  is nonincreasing we obtain  $\mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \frac{\alpha_{n+1}}{\alpha_n} \mathcal{H}_\tau(u_{\alpha_n}; g) = \mathcal{B}_\tau$ . That is,

$$\mathcal{H}_\tau(u_{\alpha_n}; g) \leq \mathcal{H}_\tau(u_{\alpha_{n+1}}; g) \leq \mathcal{B}_\tau.$$

The rest of the proof is analog to the one of Theorem 7.9.  $\square$

Nothing is known about the convergence of the APS-algorithm, if  $\mathcal{B}_\tau(\cdot)$  indeed depends on  $u$  and  $\mathcal{B}_\tau(u_{\alpha_n})$  is used instead of a fixed constant. In particular, in our numerics for some examples, in particular for the application of removing random-valued impulse noise with  $r = 0.05$ , we even observe that starting from a certain iteration the sequence  $(\alpha_n)_n$  oscillates between two states, see Fig. 7.10(c). This behavior can be attributed to the fact that, for example, if  $H_\tau(u_{\alpha_n}) \leq \mathcal{B}_\tau(u_{\alpha_n})$ , then it is not guaranteed that also  $H_\tau(u_{\alpha_{n+1}}) \leq \mathcal{B}_\tau(u_{\alpha_{n+1}})$ , which is essential for the convergence.

The second assumption in the previous theorem, i.e., the nonincrease of the function  $\alpha \mapsto \frac{\mathcal{H}_\tau(u_\alpha; g)}{\alpha}$ , can be slightly loosened, since for the convergence of the APS-algorithm it is enough to demand the nonincrease starting from a certain iteration  $\tilde{n} \geq 0$ . That is, if there exists a region  $U \subset \mathbb{R}^+$  where  $\alpha \mapsto \frac{\mathcal{H}_\tau(u_\alpha; g)}{\alpha}$  is nonincreasing and  $(\alpha_n)_{n \geq \tilde{n}} \subset U$ , then the algorithm converges; see Fig. 7.5. Analytically, this can be easily shown via Theorem 7.11 by just considering  $\alpha_{\tilde{n}}$  as the initial value of the algorithm. If  $\tau = 2$ , similar to the CPS-algorithm, we are able to show the following monotonicity property.

**Proposition 7.12.** *If there exists a constant  $c > 0$  such that  $\|T^*(Tu - g)\|_{L^2(\Omega)} = c\|Tu - g\|_{L^2(\Omega)}$  for all  $u \in L^2(\Omega)$ , then the function  $\alpha \mapsto \frac{\sqrt{\mathcal{H}_2(u_\alpha; g)}}{\alpha}$  is nonincreasing, where  $u_\alpha$  is a minimizer of  $\mathcal{J}_2(u, \alpha)$ .*

*Proof.* We start by replacing the functional  $\mathcal{J}_2$  by a family of surrogate functionals denoted by  $\bar{\mathcal{S}}$  and defined for  $u, a \in X$  as

$$\begin{aligned} \bar{\mathcal{S}}(u, a) &:= \mathcal{J}_2(u, \alpha) + \frac{\delta}{2} \|u - a\|_{L^2(\Omega)}^2 - \frac{1}{2} \|T(u - a)\|_{L^2(\Omega)}^2 \\ &= \delta \|u - z(a)\|_{L^2(\Omega)}^2 + 2\alpha \int_\Omega |Du| + \psi(a, g, T) \end{aligned}$$

where  $\delta > \|T\|^2$ ,  $z(a) := a - \frac{1}{\delta} T^*(Ta - g)$ , and  $\psi$  is a function independent of  $u$ . It can be shown that the iteration

$$u_{\alpha,0} \in X, \quad u_{\alpha,k+1} = \arg \min_u \bar{\mathcal{S}}(u, u_{\alpha,k}), \quad k \geq 0 \tag{7.18}$$

## II. Parameter Selection Methods for Total Variation Models

generates a sequence  $(u_{\alpha,k})_k$  which converges weakly for  $k \rightarrow \infty$  to a minimizer  $u_\alpha$  of  $\mathcal{J}_2(u, \alpha)$ , see for example [32]. The unique minimizer  $u_{\alpha,k+1}$  is given by  $u_{\alpha,k+1} = (I - P_{\frac{\alpha}{\delta}K})(z(u_{\alpha,k}))$ , where  $K$  is the closure of the set

$$\{\operatorname{div} \xi : \xi \in C_c^1(\Omega, \mathbb{R}^2), |\xi(x)| \leq 1 \forall x \in \Omega\}$$

and  $P_K(u) := \arg \min_{v \in K} \|u - v\|_{L^2(\Omega)}$ ; see [18]. Then for  $k \rightarrow \infty$ , let us define

$$\tilde{f}(\alpha) := \left\| P_{\frac{\alpha}{\delta}K}(z(u_\alpha)) \right\|_{L^2(\Omega)} = \left\| \frac{1}{\delta} T^*(Tu_\alpha - g) \right\|_{L^2(\Omega)}.$$

Since  $\|T^*(Tu - g)\|_{L^2(\Omega)} = c\|Tu - g\|_{L^2(\Omega)}$ , it follows that  $\tilde{f}(\alpha) = \frac{c}{\delta} \sqrt{2\mathcal{H}_2(u_\alpha; g)}$ . The assertion follows by applying [18, Lemma 4.1], which is extendable to infinite dimensions, to  $\tilde{f}$  and by noting that the nonincrease of  $\alpha \mapsto \frac{\tilde{f}(\alpha)}{\alpha}$  implies the nonincrease of  $\alpha \mapsto \frac{\sqrt{\mathcal{H}_2(u_\alpha; g)}}{\alpha}$ .  $\square$

We remark that for convolution type of operators the assumption of Proposition 7.12 does not hold in general. However, there exist several operators  $T$ , relevant in image processing, with the property  $\|T^*(Tu - g)\|_{L^2(\Omega)} = \|Tu - g\|_{L^2(\Omega)}$ . Such operators include  $T = I$  for image denoising,  $T = 1_D$  for image inpainting, where  $1_D$  denotes the characteristic function of the domain  $D \subset \Omega$ , and  $T = S \circ A$ , where  $S$  is a subsampling operator and  $A$  is an analysis operator of a Fourier or orthogonal wavelet transform. The latter type of operator is used for reconstructing signals from partial Fourier data [17] or in wavelet inpainting [25], respectively. For all such operators the function  $\alpha \mapsto \frac{\sqrt{\mathcal{H}_2(u_\alpha; g)}}{\alpha}$  is nonincreasing and hence by setting  $p = \frac{1}{2}$  fixed in the pAPS-algorithm or changing the update of  $\alpha$  in the APS-algorithm to

$$\alpha_{n+1} := \sqrt{\frac{\mathcal{B}_2}{\mathcal{H}_2(u_{\alpha_n}; g)}} \alpha_n,$$

where  $\mathcal{B}_2$  is a fixed constant, chosen according to (7.6), we obtain in these situations a convergent algorithm.

We emphasize once more, that in general the nonincrease of the function  $\alpha \mapsto \frac{\mathcal{H}_\tau(u_\alpha; g)}{\alpha}$  is not guaranteed. Nevertheless, there exists always a constant  $p \geq 0$  such that  $\alpha \mapsto \frac{(\mathcal{H}_\tau(u_\alpha; g))^p}{\alpha}$  is indeed nonincreasing. For example,  $p = \frac{1}{2}$  for operators  $T$  with the property  $\|T^*(Tu - g)\|_{L^2(\Omega)} = \|Tu - g\|_{L^2(\Omega)}$ ; cf. Proposition 7.12. In particular, one easily checks the following result.

**Proposition 7.13.** *Let  $0 < \alpha \leq \beta$ , and  $u_\alpha$  and  $u_\beta$  minimizers of  $\mathcal{J}_\tau(\cdot, \alpha)$  and  $\mathcal{J}_\tau(\cdot, \beta)$ , respectively, for  $\tau = 1, 2$ . Then  $\frac{(\mathcal{H}_\tau(u_\beta; g))^p}{\beta} \leq \frac{(\mathcal{H}_\tau(u_\alpha; g))^p}{\alpha}$  if and only if  $p \leq \frac{\ln \beta - \ln \alpha}{\ln \mathcal{H}_\tau(u_\beta; g) - \ln \mathcal{H}_\tau(u_\alpha; g)}$ .*

### 7.4. Locally Constrained TV Problem

In order to enhance image details, while preserving homogeneous regions, we formulate, as in [37, 58], a locally constrained optimization problem. That is, instead of considering (7.6) we formulate

$$\min_{u \in BV(\Omega)} \int_{\Omega} |Du| \quad \text{s.t.} \quad \int_{\Omega} w(x, y) |Tu - g|^\tau(y) dy \leq \nu_\tau \quad (7.19)$$

## 7. Automated Parameter Selection for Total Variation Minimization

for almost every  $x \in \Omega$ , where  $w$  is a normalized filter, i.e.,  $w \in L^\infty(\Omega \times \Omega)$ , and  $w \geq 0$  on  $\Omega \times \Omega$  with

$$\int_{\Omega} \int_{\Omega} w(x, y) dy dx = 1 \quad \text{and} \quad \int_{\Omega} \int_{\Omega} w(x, y) |\phi(y)|^\tau dy dx \geq \epsilon \|\phi\|_{L^\tau(\Omega)}^\tau \quad (7.20)$$

for all  $\phi \in L^\tau(\Omega)$  and for some  $\epsilon > 0$  independent of  $\phi$ ; cf. [37, 58].

### 7.4.1. Local Filtering

In practice for  $w$  we may use the mean filter together with a windowing technique, see for example [37, 58]. In order to explain the main idea we continue in a discrete setting. Let  $\Omega^h$  be a discrete image domain containing  $N_1 \times N_2$  pixels,  $N_1, N_2 \in \mathbb{N}$ , and by  $|\Omega^h| = N_1 N_2$  we denote the size of the discrete image (number of pixels). We approximate functions  $u$  by discrete functions, denoted by  $u^h$ . The considered functions spaces are  $X = \mathbb{R}^{N_1 \times N_2}$  and  $Y = X \times X$ . In what follows for all  $u^h \in X$  we use the following norms

$$\|u^h\|_\tau := \|u^h\|_{\ell^\tau(\Omega^h)} = \left( \sum_{x \in \Omega^h} |u^h(x)|^\tau \right)^{1/\tau}$$

for  $1 \leq \tau < +\infty$ . Moreover we denote by  $u_\Omega^h$  the average value of  $u^h \in X$ , i.e.,  $u_\Omega^h := \frac{1}{|\Omega^h|} \sum_{x \in \Omega^h} u^h(x)$ . The discrete gradient  $\nabla^h : X \rightarrow Y$  and the discrete divergence  $\operatorname{div}^h : Y \rightarrow X$  are defined in a standard-way by forward and backward differences such that  $\operatorname{div}^h := -(\nabla^h)^*$ ; see for example [18, 22, 55, 63]. With the above notations and definitions the discretization of the general function in (7.9) is given by

$$J_\tau(u^h, \alpha) := H_\tau(u^h) + R_\alpha(u^h) \quad (7.21)$$

where  $H_\tau(u^h) = \frac{1}{\tau} \|T^h u^h - g^h\|_\tau^\tau$ ,  $\tau \in \{1, 2\}$ ,  $T^h : X \rightarrow X$  is a bounded linear operator,  $\alpha \in (\mathbb{R}^+)^{N_1 \times N_2}$ , and

$$R_\alpha(u^h) := \sum_{x \in \Omega^h} \alpha(x) |\nabla^h u^h(x)|_{l^2} \quad (7.22)$$

with  $|y|_{l^2} = \sqrt{y_1^2 + y_2^2}$  for every  $y = (y_1, y_2) \in \mathbb{R}^2$ . In the sequel if  $\alpha$  is a scalar or  $\alpha \equiv 1$  in (7.22), we write instead of  $R_\alpha$  or  $R_1$  just  $\alpha R$  or  $R$ , respectively, i.e.,

$$R(u^h) = \sum_{x \in \Omega^h} |\nabla^h u^h(x)|_{l^2}$$

is the discrete total variation of  $u$  in  $\Omega^h$ , and we write  $\bar{E}_\tau$  instead of  $E_\tau$  to indicate that  $\alpha$  is constant. Introducing some step-size  $h$ , then for  $h \rightarrow 0$  (i.e. the number of pixels  $N_1 N_2$  goes to infinity) one can show, similar as for the case  $\alpha \equiv 1$ , that  $R_\alpha$   $\Gamma$ -converges to  $\int_{\Omega} \alpha |\nabla u|$ ; see [12, 62].

We turn now to the locally constrained minimization problem, which is given in the discrete setting as

$$\min_{u^h \in X} R(u^h) \quad \text{s.t.} \quad S_{i,j}^\tau(u^h) \leq \frac{\nu_\tau}{\tau} \quad \text{for all } x_{i,j} \in \Omega^h. \quad (7.23)$$

## II. Parameter Selection Methods for Total Variation Models

Here  $\nu_\tau$  is a fixed constant and

$$S_{i,j}^\tau(u^h) := \frac{1}{M_{i,j}} \sum_{x_{s,t} \in \mathcal{I}_{i,j}} \frac{1}{\tau} |(T^h u^h)(x_{s,t}) - g^h(x_{s,t})|^\tau$$

denotes the local residual at  $x_{i,j} \in \Omega^h$  with  $\mathcal{I}_{i,j}$  being some suitable set of pixels around  $x_{i,j}$  of size  $M_{i,j}$ , i.e.,  $M_{i,j} = |\mathcal{I}_{i,j}|$ . For example, in [37, 54, 58] for  $\mathcal{I}_{i,j}$  the set

$$\Omega_{i,j}^\omega = \left\{ x_{s+i,t+j} \in \Omega^h : -\frac{\omega-1}{2} \leq s, t \leq \frac{\omega-1}{2} \right\}$$

with a symmetric extension at the boundary and with  $\omega$  being odd is used. That is,  $\Omega_{i,j}^\omega$  is a set of pixels in a  $\omega$ -by- $\omega$  window centered at  $x_{i,j}$ , i.e.,  $M_{i,j} = \omega^2$  for all  $i, j$ , such that  $\Omega_{i,j}^\omega \not\subset \Omega^h$  for  $x_{i,j}$  sufficiently close to  $\partial\Omega$ . Additionally we denote by  $\tilde{\Omega}_{i,j}^\omega$  a set of pixels in a window centered at  $x_{i,j}$  without any extension at the boundary, i.e.,

$$\tilde{\Omega}_{i,j}^\omega = \left\{ x_{s+i,t+j} : \max \left\{ 1 - (i, j), -\frac{\omega-1}{2} \right\} \leq (s, t) \leq \min \left\{ \frac{\omega-1}{2}, (N_1 - i, N_2 - j) \right\} \right\}.$$

Hence  $\tilde{\Omega}_{i,j}^\omega \subset \Omega^h$  for all  $x_{i,j} \in \Omega^h$ . Before we analyze the difference between  $\Omega_{i,j}^\omega$  and  $\tilde{\Omega}_{i,j}^\omega$  with respect to the constrained minimization problem (7.23), we note that, since  $T^h$  does not annihilate constant functions, the existence of a solution of (7.23) is guaranteed; see [37, Theorem 2][58, Theorem 2].

In the following we set  $B_\tau := \frac{\nu_\tau}{\tau} |\Omega^h|$ .

**Proposition 7.14.** (i) If  $u^h$  is a solution of (7.23) with  $\mathcal{I}_{i,j} = \Omega_{i,j}^\omega$ , then  $H_\tau(u^h) < B_\tau$ .

(ii) If  $u$  is a solution of (7.23) with  $\mathcal{I}_{i,j} = \tilde{\Omega}_{i,j}^\omega$ , then  $H_\tau(u^h) \leq B_\tau$ .

*Proof.* (i) Since  $u^h$  is a solution of (7.23) and  $\Omega_{i,j}^\omega$  is a set of pixels in a  $\omega$ -by- $\omega$  window, we have

$$\begin{aligned} B_\tau &\geq \sum_{i,j} S_{i,j}^\tau(u^h) = \sum_{i,j} \frac{1}{\tau \omega^2} \sum_{x_{s,t} \in \Omega_{i,j}^\omega} |g^h(x_{s,t}) - T^h u^h(x_{s,t})|^\tau \\ &> \frac{1}{\tau} \sum_{i,j} |g^h(x_{i,j}) - T^h u^h(x_{i,j})|^\tau = H_\tau(u^h). \end{aligned}$$

Here we used that due to the sum over  $i, j$  each element (pixel) in  $\Omega_{i,j}^\omega$  appears at most  $\omega^2$  times. More precisely, any pixel-coordinate in the set  $\Lambda^\omega := \{(i, j) : \min\{i-1, j-1, N_1 - i, N_2 - j\} \geq \frac{\omega-1}{2}\}$  occurs exactly  $\omega^2$ -times, while any other pixel-coordinate appears strictly less than  $\omega^2$ -times. This shows the first statement.

(ii) For a minimizer  $u^h$  of (7.23) we obtain

$$\begin{aligned} B_\tau &\geq \sum_{i,j} S_{i,j}^\tau(u^h) = \sum_{i,j} \frac{1}{\tau M_{i,j}} \sum_{x_{s,t} \in \Omega_{i,j}^\omega} |g^h(x_{s,t}) - T^h u^h(x_{s,t})|^\tau \\ &= \frac{1}{\tau} \sum_{i,j} |g^h(x_{i,j}) - T^h u^h(x_{i,j})|^\tau = H_\tau(u^h), \end{aligned}$$

which concludes the proof. □

## 7. Automated Parameter Selection for Total Variation Minimization

Note, that if  $\mathcal{I}_{i,j} = \tilde{\Omega}_{i,j}^\omega$  then by Proposition 7.14 a minimizer of (7.23) also satisfies the constraint of the problem

$$\min_{u^h \in X} R(u^h) \quad \text{s.t.} \quad H_\tau(u^h) \leq B_\tau \quad (7.24)$$

(discrete version of (7.11)) but is in general of course not a solution of (7.24).

**Proposition 7.15.** *Let  $\mathcal{I}_{i,j} = \tilde{\Omega}_{i,j}^\omega$ ,  $u_s^h$  be a minimizer of (7.24) and  $u_l^h$  be a minimizer of (7.23), then  $R(u_s^h) \leq R(u_l^h)$ .*

*Proof.* Assume that  $R(u_s^h) > R(u_l^h)$ . Since  $u_s^h$  is a solution of (7.24) it satisfies the constraint  $H_\tau(u_s^h) \leq B_\tau$ . By Proposition 7.14 we also have  $H_\tau(u_l^h) \leq B_\tau$ . Since  $R(u_s^h) > R(u_l^h)$ ,  $u_s^h$  is not the solution of (7.24) which is a contradiction. Hence,  $R(u_s^h) \leq R(u_l^h)$ .  $\square$

**Remark 7.16.** *Proposition 7.14 and its consequence are not special properties of the discrete setting. Let the filter  $w$  in (7.19) be such that the inequality in (7.20) becomes an equality with  $\epsilon = 1/|\Omega|$ , as it is the case in Proposition 7.14(ii), then a solution  $u_l$  of the locally constrained minimization problem (7.19) satisfies*

$$\mathcal{H}_\tau(u_l; g) \leq \frac{\nu_\tau}{\tau} |\Omega| \quad \text{and} \quad \int_\Omega |Du_l| \geq \int_\Omega |Du_s|$$

where  $u_s$  is a solution of (7.11).

From Proposition 7.15 and Remark 7.16 we conclude, since  $R(u_s^h) \leq R(u_l^h)$  and  $\int_\Omega |Du_s| \leq \int_\Omega |Du_l|$ , that  $u_s^h$  and  $u_s$  are smoother than  $u_l^h$  and  $u_l$ , respectively. Hence the solution of the locally constrained minimization problem is expected to preserve details better than the minimizer of the globally constrained optimization problem. Since noise can be interpreted as fine details, which we actually want to eliminate, this could also mean, that noise is possibly left in the image.

### 7.4.2. Locally Adaptive Total Variation Algorithm

Whenever  $\nu_\tau$  depends on  $\hat{u}$  problem (7.23) results in a quite nonlinear problem. Instead of considering nonlinear constraints we choose as in Section 7.3 a reference image  $\tilde{u}$  and compute an approximate  $\nu_\tau = \nu_\tau(\tilde{u})$ . Note, that in our discrete setting for salt-and-pepper noise we have now

$$\nu_1(u^h) := r_2 - (r_2 - r_1) \frac{1}{|\Omega|} \sum_{x \in \Omega^h} (T^h u^h)(x)$$

and for random-valued impulse noise we have

$$\nu_1(u^h) := \frac{1}{|\Omega^h|} \sum_{x \in \Omega^h} r \left( (T^h u^h)(x)^2 - (T^h u^h)(x) + \frac{1}{2} \right).$$

In our below proposed locally adaptive algorithms we choose as a reference image the current approximation (see LATV- and pLATV-algorithm below), as also done in the pAPS- and APS-algorithm above. Then we are seeking for a solution  $u^h$  such that  $S_{i,j}^\tau(u^h)$  is close to  $\frac{\nu_\tau}{\tau}$ .

We note, that for large  $\alpha > 0$  the minimization of (7.21) yields an over-smoothed restoration  $u_\alpha^h$  and the residual contains details, i.e., we expect  $H_\tau(u_\alpha^h) > B_\tau$ . Hence, if  $S_{i,j}^\tau(u_\alpha^h) > \frac{\nu_\tau}{\tau}$  we suppose that this is due to image details contained in the local residual image. In this

## II. Parameter Selection Methods for Total Variation Models

situation we intend to decrease  $\alpha$  in the local regions  $\mathcal{I}_{i,j}$ . In particular, we define, similar as in [37, 58], the local quantity  $f_{i,j}^\omega$  by

$$f_{i,j}^\omega := \begin{cases} S_{i,j}^\tau(u_\alpha^h) & \text{if } S_{i,j}^\tau(u_\alpha^h) > \frac{\nu_\tau}{\tau}, \\ \frac{\nu_\tau}{\tau} & \text{otherwise.} \end{cases}$$

Note, that  $\frac{\nu_\tau}{\tau f_{i,j}^\omega} \leq 1$  for all  $i, j$  and hence we set

$$\alpha(x_{i,j}) := \frac{1}{M_{i,j}} \sum_{x_{s,t} \in \mathcal{I}_{i,j}} \left( \frac{\nu_\tau}{\tau f_{s,t}^\omega} \right)^p \alpha(x_{s,t}). \quad (7.25)$$

On the other hand, for small  $\alpha > 0$  we get an under-smoothed image  $u_\alpha^h$ , which still contains noise, i.e., we expect  $H_\tau(u_\alpha^h) < B_\tau$ . Analogously, if  $S_{i,j}^\tau(u_\alpha^h) \leq \frac{\nu_\tau}{\tau}$ , we suppose that there is still noise left outside the residual image in  $\mathcal{I}_{i,j}$ . Hence we intend to increase  $\alpha$  in the local regions  $\mathcal{I}_{i,j}$  by defining

$$f_{i,j}^\omega := \begin{cases} S_{i,j}^\tau(u_\alpha^h) & \text{if } S_{i,j}^\tau(u_\alpha^h) < \frac{\nu_\tau}{\tau}, \\ \frac{\nu_\tau}{\tau} & \text{otherwise,} \end{cases}$$

and setting  $\alpha$  as in (7.25). Notice, that now  $\frac{\nu_\tau}{\tau f_{i,j}^\omega} \geq 1$ . These considerations lead to the following locally adapted total variation algorithm.

**LATV-algorithm:** Choose  $\alpha_0 > 0$ ,  $p := p_0 > 0$ , and set  $n := 0$ .

1) Compute  $u_{\alpha_n}^h \in \arg \min_{u^h \in X} J_\tau(u^h, \alpha_n)$

2) (a) If  $H_\tau(u_{\alpha_0}^h) > B_\tau(u_{\alpha_0}^h)$ , then set

$$f_{i,j}^\omega := \max \left\{ S_{i,j}^\tau(u_{\alpha_n}^h), \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau} \right\}$$

(b) If  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$ , then set

$$f_{i,j}^\omega := \max \left\{ \min \left\{ S_{i,j}^\tau(u_{\alpha_n}^h), \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau} \right\}, \varepsilon \right\}$$

3) Update

$$\alpha_{n+1}(x_{i,j}) := \frac{1}{M_{i,j}} \sum_{x_{s,t} \in \mathcal{I}_{i,j}} \left( \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau f_{s,t}^\omega} \right)^p \alpha_n(x_{s,t}).$$

4) Stop or set  $n := n + 1$  and return to step 1).

Here and below  $\varepsilon > 0$  is a small constant (e.g., in our experiments we choose  $\varepsilon = 10^{-14}$ ) to ensure that  $f_{i,j}^\omega > 0$ , since it may happen that  $S_{i,j}^\tau(u_{\alpha_n}^h) = 0$ .

If  $H_\tau(u_{\alpha_0}^h) > B_\tau(u_{\alpha_0}^h)$ , we stop the algorithm as soon as the residual  $H_\tau(u_{\alpha_n}^h) < B_\tau(u_{\alpha_n}^h)$  for the first time and set the desired locally varying  $\alpha^* = \alpha_n$ . If  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$ , we stop the algorithm as soon as the residual  $H_\tau(u_{\alpha_n}^h) > B_\tau(u_{\alpha_n}^h)$  for the first time and set the desired locally varying  $\alpha^* = \alpha_{n-1}$ , since  $H_\tau(u_{\alpha_{n-1}}^h) \leq \frac{\nu_\tau(u_{\alpha_{n-1}}^h)}{\tau}$ .

The LATV-algorithm has the following monotonicity properties with respect to  $(\alpha_n)_n$ .

## 7. Automated Parameter Selection for Total Variation Minimization

**Proposition 7.17.** Assume  $\mathcal{I}_{i,j} = \Omega_{i,j}^\omega$  and let  $\varepsilon > 0$  be sufficiently small. If  $\alpha_0 > 0$  such that  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$ , then the LATV-algorithm generates a sequence  $(\alpha_n)_n$  such that

$$\sum_{i,j} \alpha_{n+1}(x_{i,j}) > \sum_{i,j} \alpha_n(x_{i,j}).$$

*Proof.* By the same argument as in the proof of Proposition 7.14 we obtain

$$\sum_{i,j} \alpha_{n+1}(x_{i,j}) = \sum_{i,j} \left( \frac{(\nu_\tau(u_{\alpha_n}^h))^p}{\tau^p \omega^2} \sum_{(s,t) \in \Omega_{i,j}^\omega} \frac{\alpha_n(x_{s,t})}{(f_{s,t}^\omega)^p} \right) > \sum_{i,j} \left( \frac{(\nu_\tau(u_{\alpha_n}^h))^p \omega^2}{\tau^p \omega^2} \frac{\alpha_n(x_{i,j})}{(f_{i,j}^\omega)^p} \right).$$

Note that  $\nu_\tau(\cdot)$  is bounded from below, see Section 7.3.1. Consequently there exists an  $\varepsilon > 0$  such that  $\frac{\nu_\tau(u^h)}{\tau} \geq \varepsilon$  for any  $u^h$ . Then, since  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$  we have by the LATV-algorithm that

$$f_{i,j}^\omega := \max \left\{ \min \left\{ S_{i,j}^\tau(u_{\alpha_n}^h), \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau} \right\}, \varepsilon \right\} \leq \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau}$$

and hence  $\sum_{i,j} (\alpha_{n+1})(x_{i,j}) > \sum_{i,j} (\alpha_n)(x_{i,j})$ .  $\square$

**Proposition 7.18.** Let  $\mathcal{I}_{i,j} = \tilde{\Omega}_{i,j}^\omega$  and  $\varepsilon > 0$  be sufficiently small.

(i) If  $\alpha_0 > 0$  such that  $H_\tau(u_{\alpha_0}^h) > B_\tau(u_{\alpha_0}^h)$ , then the LATV-algorithm generates a sequence  $(\alpha_n)_n$  such that

$$\sum_{i,j} (\alpha_{n+1})(x_{i,j}) \leq \sum_{i,j} (\alpha_n)(x_{i,j}).$$

(ii) If  $\alpha_0 > 0$  such that  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$ , then the LATV-algorithm generates a sequence  $(\alpha_n)_n$  such that

$$\sum_{i,j} (\alpha_{n+1})(x_{i,j}) \geq \sum_{i,j} (\alpha_n)(x_{i,j}).$$

*Proof.* (i) By the same argument as in the proof of Proposition 7.14 and since  $f_{i,j}^\omega := \max \left\{ S_{i,j}^\tau(u_{\alpha_n}^h), \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau} \right\} \geq \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau}$  we obtain

$$\begin{aligned} \sum_{i,j} \alpha_{n+1}(x_{i,j}) &= \sum_{i,j} \left( \frac{\nu_\tau(u_{\alpha_n}^h)^p}{\tau^p M_{i,j}} \sum_{x_{s,t} \in \tilde{\Omega}_{i,j}^\omega} \frac{\alpha_n(x_{s,t})}{(f_{s,t}^\omega)^p} \right) \\ &= \sum_{i,j} \left( \left( \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau} \right)^p \frac{\alpha_n(x_{i,j})}{(f_{i,j}^\omega)^p} \right) \leq \sum_{i,j} \alpha_n(x_{i,j}). \end{aligned}$$

(ii) Since  $\nu_\tau(\cdot)$  is bounded from below, see Section 7.3.1, there exists an  $\varepsilon > 0$  such that  $\frac{\nu_\tau(u^h)}{\tau} \geq \varepsilon$  for any  $u^h$ . Hence

$$f_{i,j}^\omega := \max \left\{ \min \left\{ S_{i,j}^\tau(u_{\alpha_n}^h), \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau} \right\}, \varepsilon \right\} \leq \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau}$$

## II. Parameter Selection Methods for Total Variation Models

and by the same arguments as above we get

$$\sum_{i,j} \alpha_{n+1}(x_{i,j}) = \sum_{i,j} \left( \left( \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau} \right)^p \frac{\alpha_n(x_{i,j})}{(f_{i,j}^\omega)^p} \right) \geq \sum_{i,j} \alpha_n(x_{i,j}).$$

□

In contrast to the pAPS-algorithm in the LATV-algorithm the power  $p > 0$  is not changed during the iterations and should be chosen sufficiently small, e.g., we set  $p = \frac{1}{2}$  in our experiments. Note, that small  $p$  only allow small changes of  $\alpha$  in each iteration. In this way the algorithm is able to generate a function  $\alpha^*$  such that  $H_\tau(u_{\alpha^*}^h)$  is very close to  $\frac{\nu_\tau(u_{\alpha^*}^h)}{\tau}$ . On the contrary, small  $p$  have the drawback that the number of iterations till termination are kept large. Since the parameter  $p$  has to be chosen manually, the LATV-algorithm, at least in the spirit, seems to be similar to Uzawa's method, where also a parameter has to be chosen. The proper choice of such a parameter might be complicated and hence we are desiring for an algorithm where we do not have to tune parameters manually. Because of this and motivated by the pAPS-algorithm we propose the following  $p$  adaptive algorithm:

**pLATV-algorithm:** Choose  $\alpha_0 > 0$ ,  $p := p_0 > 0$ , and set  $n := 0$ .

0) Compute  $u_{\alpha_n} \in \arg \min_{u^h \in X} J_\tau(u^h, \alpha_n)$

1) (a) If  $H_\tau(u_{\alpha_0}^h) > B_\tau(u_{\alpha_0}^h)$ , then set

$$f_{i,j}^\omega := \max \left\{ S_{i,j}^\tau(u_{\alpha_n}^h), \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau} \right\}$$

(b) If  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$ , then set

$$f_{i,j}^\omega := \max \left\{ \min \left\{ S_{i,j}^\tau(u_{\alpha_n}^h), \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau} \right\}, \varepsilon \right\}$$

2) Update

$$\alpha_{n+1}(x_{i,j}) := \frac{\alpha_n(x_{i,j})}{M_{i,j}} \sum_{x_{s,t} \in \mathcal{I}_{i,j}} \left( \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau f_{s,t}^\omega} \right)^p.$$

3) Compute  $u_{\alpha_{n+1}}^h \in \arg \min_{u^h \in X} J_\tau(u^h, \alpha_{n+1})$

4) (a) if  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$

(i) if  $H_\tau(u_{\alpha_{n+1}}^h) \leq B_\tau(u_{\alpha_{n+1}}^h)$ , go to step 5)

(ii) if  $H_\tau(u_{\alpha_{n+1}}^h) > B_\tau(u_{\alpha_{n+1}}^h)$ , decrease  $p$ , e.g., set  $p = p/10$ , and go to step 2)

(b) if  $H_\tau(u_{\alpha_0}^h) > B_\tau(u_{\alpha_0}^h)$

(i) if  $H_\tau(u_{\alpha_{n+1}}^h) \geq B_\tau(u_{\alpha_{n+1}}^h)$ , go to step 5)

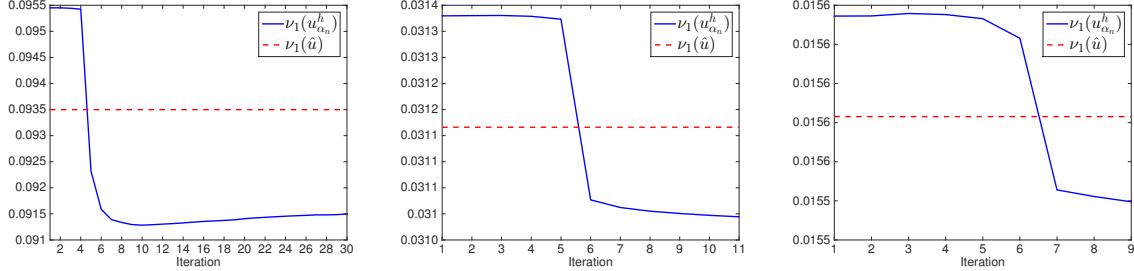
(ii) if  $H_\tau(u_{\alpha_{n+1}}^h) < B_\tau(u_{\alpha_{n+1}}^h)$ , decrease  $p$ , e.g., set  $p = p/10$ , and go to step 2)

5) Stop or set  $n := n + 1$  and return to step 1).

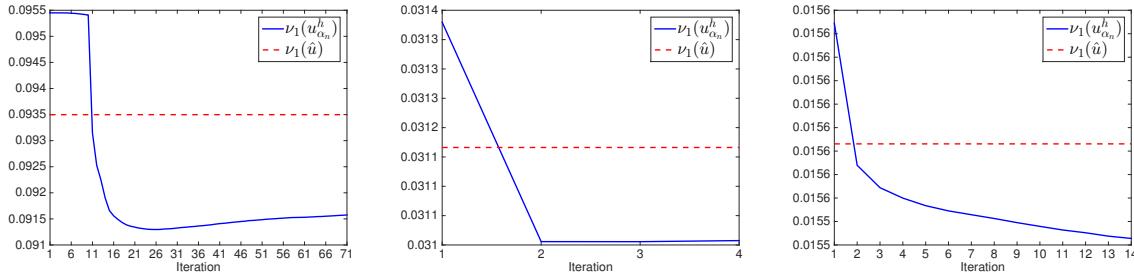
In our numerical experiments this algorithm is terminated as soon as  $|H_\tau(u_{\alpha_n}^h) - B_\tau(u_{\alpha_n}^h)| \leq 10^{-6}$  and  $H_\tau(u_{\alpha_n}^h) \leq B_\tau(u_{\alpha_n}^h)$ . Additionally we stop iterating when  $p$  is less than machine

## 7. Automated Parameter Selection for Total Variation Minimization

precision, since then anyway no progress is to expect. Due to the adaptive choice of  $p$  we obtain a monotonic behavior of the sequence  $(\alpha_n)_n$ .



**Figure 7.2:** Progress of  $\nu_\tau(u_{\alpha_n}^h)$  of the LATV-algorithm with  $p = \frac{1}{8}$  and  $\alpha_0 = 10^{-2}$  for removing random-valued impulse noise with  $r = 0.3$  (left),  $r = 0.1$  (middle),  $r = 0.05$  (right) from the cameraman-image (cf. Fig. 7.4(b)).



**Figure 7.3:** Progress of  $\nu_\tau(u_{\alpha_n}^h)$  of the pLATV-algorithm with  $p_0 = \frac{1}{2}$  and  $\alpha_0 = 10^{-2}$  for removing random-valued impulse noise with  $r = 0.3$  (left),  $r = 0.1$  (middle),  $r = 0.05$  (right) from the cameraman-image (cf. Fig. 7.4(b)).

**Proposition 7.19.** *The sequence  $(\alpha_n)_n$  generated by the pLATV-algorithm is for any point  $x \in \Omega$  monotone. In particular, it is monotonically decreasing for  $\alpha_0$  such that  $H_\tau(u_{\alpha_0}^h) > B_\tau(u_{\alpha_0}^h)$ , and monotonically increasing for  $\alpha_0$  such that  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$ .*

*Proof.* For  $H_\tau(u_{\alpha_0}^h) > B_\tau(u_{\alpha_0}^h)$  we can show by induction that by the pLATV-algorithm  $f_{i,j}^\omega \geq \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau}$  and hence  $1 \geq \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau f_{i,j}^\omega}$  for all  $n$ . Then by the definition of  $\alpha_{n+1}$  it follows

$$\begin{aligned} \alpha_{n+1}(x_{i,j}) &:= \frac{\alpha_n(x_{i,j})}{M_{i,j}} \sum_{x_{s,t} \in \mathcal{I}_{i,j}} \left( \frac{\nu_\tau(u_{\alpha_n}^h)}{\tau f_{s,t}^\omega} \right)^p \\ &\leq \alpha_n(x_{i,j}). \end{aligned}$$

By similar arguments we obtain for  $\alpha_0$  with  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$  that  $\alpha_{n+1}(x_{i,j}) \geq \alpha_n(x_{i,j})$  for all  $x_{i,j} \in \Omega$ .  $\square$

We are aware of the fact that using  $u_{\alpha_n}^h$  as a reference image in the LATV- and pLATV-algorithm to compute  $\nu_\tau$  may commit errors. However, we recall that for Gaussian noise we

## II. Parameter Selection Methods for Total Variation Models

set  $\nu_2 = \sigma^2$  and for salt-and-pepper noise with  $r_1 = r_2$  we have  $\nu_1 = r_1$ . In these cases  $\nu_\tau$  does not depend on the original image and hence we do not commit any error by computing  $\nu_\tau$ . For random-valued impulse noise corrupted images the situation is different and  $\nu_1$  indeed depends on the true image. In this situation errors may be committed when  $u_{\alpha_n}^h$  is used as a reference image for calculating  $\nu_\tau$ ; see Figs. 7.2 and 7.3. Hence, in order to improve the proposed algorithm, for such cases for future research it might be of interest to find the optimal reference image to obtain a good approximation of the real value  $\nu_\tau$ .

In contrast to the SA-TV algorithm presented in [37, 58], where the initial regularization parameter has to be chosen sufficiently small, in the LATV-algorithm as well as in the pLATV-algorithm the initial value  $\alpha_0$  can be chosen arbitrarily positive. However, in the case  $H_\tau(u_{\alpha_0}^h) > B_\tau(u_{\alpha_0}^h)$  we cannot guarantee in general that the solution  $u_\alpha$  obtained by the pLATV-algorithm fulfills  $H_\tau(u_\alpha^h) \leq B_\tau(u_\alpha^h)$ , not even if  $B_\tau(\cdot)$  is constant, due to the stopping criterion with respect to the power  $p$ . On the contrary, if  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$ , then the pLATV-algorithm generates a sequence  $(u_{\alpha_n}^h)_n$  such that  $H_\tau(u_{\alpha_n}^h) \leq B_\tau(u_{\alpha_n}^h)$  for all  $n$  and hence also for the solution of the algorithm. As a consequence we would wish to choose  $\alpha_0 > 0$  such that  $H_\tau(u_{\alpha_0}^h) \leq B_\tau(u_{\alpha_0}^h)$ , which may be realized by the following simple automated procedure:

**Algorithm 1:** Input:  $\alpha_0 > 0$  (arbitrary);

- 1) Compute  $u_{\alpha_0}^h \in \arg \min_{u^h \in X} J_\tau(u^h, \alpha_0)$ .
- 2) If  $H_\tau(u_{\alpha_0}^h) > B_\tau(u_{\alpha_0}^h)$  decrease  $\alpha_0$  by setting  $\alpha_0 = c_{\alpha_0} \alpha_0$ , with  $c_{\alpha_0} \in (0, 1)$ , and continue with step 1), otherwise stop and return  $\alpha_0$ .

## 7.5. Total Variation Minimization

In this section we are concerned with developing numerical methods for computing a minimizer of the discrete multiscale  $L^\tau$ -TV model, i.e.,

$$\min_{u^h \in X} \{J_\tau(u^h, \alpha) := H_\tau(u^h) + R_\alpha(u^h)\}. \quad (7.26)$$

### 7.5.1. $L^2$ -TV Minimization

Here we consider the case  $\tau = 2$ , i.e., the minimization problem

$$\min_{u^h \in X} \frac{1}{2} \|T^h u^h - g^h\|_2^2 + R_\alpha(u^h), \quad (7.27)$$

and present solution methods, first for the case  $T^h = I$  and then for general linear bounded operators  $T^h$ .

#### 7.5.1.1. An Algorithm for Image Denoising

If  $T^h = I$ , then (7.27) becomes an image denoising problem, i.e., the minimization problem

$$\min_{u^h \in X} \|u^h - g^h\|_2^2 + 2R_\alpha(u^h). \quad (7.28)$$

For solving this problem we use the algorithm of Chambolle and Pock [22], which leads to the following iterative scheme:

## 7. Automated Parameter Selection for Total Variation Minimization

**Chambolle-Pock algorithm:** Initialize  $\tau, \sigma > 0$ ,  $\theta \in [0, 1]$ ,  $(\mathbf{p}_0^h, u_0^h) \in Y \times X$ , set  $\bar{u}_0^h = u_0^h$ , and set  $n = 0$ .

1. Compute

$$\mathbf{p}_{n+1}^h(x) = \frac{\mathbf{p}_n^h(x) + \sigma \nabla^h \bar{u}_n^h(x)}{\max \left\{ \frac{1}{\alpha(x)} |\mathbf{p}_n^h(x) + \sigma \nabla^h \bar{u}_n^h(x)|, 1 \right\}},$$

for all  $x \in \Omega^h$ .

2. Compute  $u_{n+1}^h = \frac{u_n^h + \tau \operatorname{div}^h \mathbf{p}_{n+1}^h + \tau g^h}{1 + \tau}$ .

3. Set  $\bar{u}_{n+1}^h = u_{n+1}^h + \theta(u_{n+1}^h - u_n^h)$ .

4. Stop or set  $n := n + 1$  and return to step 1).

In our numerical experiments we choose  $\theta = 1$ . In particular, in [22] it is shown that for  $\theta = 1$  and  $\tau\sigma\|\nabla^h\|^2 < 1$  the algorithm converges.

### 7.5.1.2. An Algorithm for Linear Bounded Operators

Assume, that  $T^h$  is a linear bounded operator from  $X$  to  $X$ , different to the identity  $I$ . Then instead of minimizing (7.27) directly, we introduce the surrogate functional

$$\begin{aligned} \mathcal{S}(u^h, a^h) &:= \frac{1}{2} \|T^h u^h - g^h\|_2^2 + R_\alpha(u^h) + \frac{\delta}{2} \|u^h - a^h\|_2^2 - \frac{1}{2} \|T^h(u^h - a^h)\|_2^2 \\ &= \frac{\delta}{2} \|u^h - z(a^h)\|_2^2 + R_\alpha(u^h) + \psi(a^h, g^h, T^h), \end{aligned} \quad (7.29)$$

with  $a^h, u^h \in X$ ,  $z(a^h) = a^h - \frac{1}{\delta}(T^h)^*(T^h a^h - g^h)$ ,  $\psi$  a function independent of  $u^h$ , and where we assume  $\delta > \|T^h\|^2$ ; see [31, 32]. Note that

$$\min_{u^h \in X} \mathcal{S}(u^h, a^h) \Leftrightarrow \min_{u \in X} \|u^h - z(a^h)\|_2^2 + 2R_{\frac{\alpha}{\delta}}(u^h)$$

and hence to obtain a minimizer amounts to solve a minimization problem of the type (7.28) and can be solved as described in Section 7.5.1.1. Then an approximate solution of (7.27) can be computed by the following iterative algorithm: Choose  $u_0^h \in X$  and iterate for  $n \geq 0$

$$u_{n+1}^h = \arg \min_{u^h \in X} \mathcal{S}(u^h, u_n^h). \quad (7.30)$$

For scalar  $\alpha$  it is shown in [28, 31, 32] that this iterative procedure generates a sequence  $(u_n^h)_n$  which converges to a minimizer of (7.27). This convergence property can be easily extended to our nonscalar case yielding the following result.

**Theorem 7.20.** *For  $\alpha : \Omega \rightarrow \mathbb{R}^+$  the scheme in (7.30) generates a sequence  $(u_n^h)_n$ , which converges to a solution of (7.27) for any initial choice of  $u_0^h \in X$ .*

*Proof.* A proof can be accomplished analogue to [32]. □

## II. Parameter Selection Methods for Total Variation Models

### 7.5.2. An Algorithm for $L^1$ -TV Minimization

The computation of a minimizer of

$$\min_{u^h \in X} \|T^h u^h - g^h\|_1 + R_\alpha(u^h), \quad (7.31)$$

is due to the nonsmooth  $\ell^1$ -term in general more complicated than obtaining a solution of the  $L^2$ -TV model. Here we suggest to employ a trick, proposed in [5] for  $L^1$ -TV minimization problems with a scalar regularization parameter, to solve (7.31) in two steps. In particular, we substitute the argument of the  $\ell^1$ -norm by a new variable  $v$ , penalize the functional by an  $L^2$ -term, which should keep the difference between  $v$  and  $Tu - g$  small, and minimize with respect to  $v$  and  $u$ . That is, we replace the original minimization (7.31) by

$$\min_{v^h, u^h \in X} \|v^h\|_1 + \frac{1}{2\gamma} \|T^h u^h - g^h - v^h\|_2^2 + R_\alpha(u^h), \quad (7.32)$$

where  $\gamma > 0$  is small, so that we have  $g^h \approx T^h u^h - v^h$ . Actually, it can be shown that (7.32) converges to (7.31) as  $\gamma \rightarrow 0$ . In our experiments we actually choose  $\gamma = 10^{-2}$ . This leads to the following alternating algorithm.

**$L^1$ -TV $_\alpha$  algorithm:** Initialize  $\alpha > 0$ ,  $u_0^h \in X$  and set  $n := 0$ .

1) Compute

$$v_{n+1}^h = \arg \min_{v^h \in X} \|v^h\|_1 + \frac{1}{2\gamma} \|T^h u_n^h - g^h - v^h\|_2^2$$

2) Compute

$$u_{n+1}^h \in \arg \min_{u^h \in X} \frac{1}{2\gamma} \|T^h u^h - g^h - v_{n+1}^h\|_2^2 + R_\alpha(u^h)$$

3) Stop or set  $n := n + 1$  and return to step 1).

The minimizer  $v_{n+1}^h$  in step 1) of the  $L^1$ -TV $_\alpha$  algorithm can be easily computed via a soft-thresholding, i.e.,  $v_{n+1}^h = \text{ST}(T^h u_n^h - g^h, \gamma)$ , where

$$\text{ST}(g^h, \gamma)(x) = \begin{cases} g^h(x) - \gamma & \text{if } g^h(x) > \gamma, \\ 0 & \text{if } |g^h(x)| \leq \gamma, \\ g^h(x) + \gamma & \text{if } g^h(x) < -\gamma \end{cases}$$

for all  $x \in \Omega^h$ . The minimization problem in step 2) is equivalent to

$$\arg \min_{u^h \in X} \frac{1}{2} \|T^h u^h - g^h - v_{n+1}^h\|_2^2 + R_{\gamma\alpha}(u^h) \quad (7.33)$$

and hence is of the type (7.27). Thus an approximate solution of (7.33) can be computed as described above; see Section 7.5.1.

**Theorem 7.21.** *The sequence  $(u_n^h, v_n^h)_n$  generated by the  $L^1$ -TV $_\alpha$  algorithm converges to a minimizer of (7.32).*

*Proof.* The statement can be shown analogue to [5]. □

### 7.5.3. A Primal-Dual Method for $L^1$ -TV Minimization

For solving (7.6) with  $\tau = 1$  we suggest, alternatively to the above method, to use the primal-dual method of [58] adapted to our setting, where a Huber regularisation of the gradient of  $u$  is considered; see [58] for more details. Denoting by  $\bar{u}$  a corresponding solution of the primal problem and  $\bar{\mathbf{p}}$  the solution of the associated dual problem, the optimality conditions due to the Fenchel theorem [39] are given by

$$\begin{aligned} -\operatorname{div} \bar{\mathbf{p}}(x) &= -\kappa \Delta \bar{u}(x) + \frac{1}{\beta + \mu} T^*(T\bar{u}(x) - g(x)) + \frac{\mu}{\beta + \mu} T^* \frac{T\bar{u}(x) - g(x)}{\max\{\beta, |T\bar{u}(x) - g(x)|\}} \\ -\bar{\mathbf{p}}(x) &= \frac{1}{\gamma} \nabla \bar{u}(x) \quad \text{if } |\bar{\mathbf{p}}(x)|_{l^2} < \alpha(x) \\ -\bar{\mathbf{p}}(x) &= \alpha(x) \frac{\nabla \bar{u}(x)}{|\nabla \bar{u}(x)|_{l^2}} \quad \text{if } |\bar{\mathbf{p}}(x)|_{l^2} = \alpha(x), \end{aligned}$$

for all  $x \in \Omega$ , where  $\kappa, \beta, \mu$ , and  $\gamma$  are fixed positive constants. The latter two conditions can be summarized to  $-\bar{\mathbf{p}}(x) = \frac{\alpha(x) \nabla \bar{u}(x)}{\max\{\gamma \alpha(x), |\nabla \bar{u}(x)|_{l^2}\}}$ . Then setting  $\bar{\mathbf{q}} = -\bar{\mathbf{p}}$  and  $\bar{v} = \frac{T\bar{u} - g}{\max\{\beta, |T\bar{u} - g|\}}$  leads to the following system of equation:

$$\begin{aligned} 0 &= -\max\{\beta, |T\bar{u}(x) - g(x)|\} \bar{v} + T\bar{u}(x) - g(x) \\ 0 &= \operatorname{div} \bar{\mathbf{q}}(x) + \kappa \Delta \bar{u}(x) - \frac{1}{\beta + \mu} T^*(T\bar{u}(x) - g(x)) - \frac{\mu}{\beta + \mu} T^* \bar{v}(x) \\ 0 &= \max\{\gamma \alpha(x), |\nabla \bar{u}(x)|_{l^2}\} \bar{\mathbf{q}}(x) - \alpha(x) \nabla \bar{u}(x) \end{aligned} \quad (7.34)$$

for all  $x \in \Omega$ . This system can be solved efficiently by a semismooth Newton algorithm; see Section 7.5.3.1 for a description of the method and for the choice of the parameters  $\kappa, \beta, \mu$ , and  $\gamma$ .

Note, that different algorithms presented in the literature can also be adjusted to the case of a locally varying regularization parameter, such as [18, 21, 68]. However, it is not the scope of this paper to compare different algorithms in order to detect the most efficient one, although this is an interesting research topic in its own right.

#### 7.5.3.1. Semismooth Newton Method for Solving (7.34)

A semismooth Newton algorithm for solving (7.34) can be derived similar as in [58] by means of vector-valued variables. Therefore let  $u^h \in \mathbb{R}^N$ ,  $q^h \in \mathbb{R}^{2N}$ ,  $\alpha^h \in \mathbb{R}^N$ ,  $g^h \in \mathbb{R}^N$  where  $N = N_1 N_2$ , denote the discrete image intensity, the dual variable, the spatially dependent regularization parameter, and the observed data vector, respectively. Correspondingly we define  $\nabla^h \in \mathbb{R}^{2N \times N}$  as the discrete gradient operator,  $\Delta^h \in \mathbb{R}^{N \times N}$  as the discrete Laplace operator,  $T^h \in \mathbb{R}^{N \times N}$  as the discrete operator, and  $(T^h)^t$  as the transpose of  $T^h$ . Here  $|\cdot|$ ,  $\max\{\cdot, \cdot\}$ , and  $\operatorname{sign}(\cdot)$  are understood for vectors in a componentwise sense. Moreover, we use the function  $[\cdot] : \mathbb{R}^{2N} \rightarrow \mathbb{R}^{2N}$  with  $[\cdot]_i = [\cdot]_{i+N} = \sqrt{(v_i^h)^2 + (v_{i+N}^h)^2}$  for  $1 \leq i \leq N$ .

For solving (7.34) in every step of our Newton method we need to solve

$$\begin{pmatrix} A_k^h & -D^h(m_{\beta_k}) & 0 \\ -\frac{1}{\beta+\mu} (T^h)^t T^h + \kappa \Delta^h & -\frac{\mu}{\mu+\beta} (T^h)^t & -(\nabla^h)^t \\ B_k^h & 0 & D^h(m_{\gamma_k}) \end{pmatrix} \begin{pmatrix} \delta_u \\ \delta_v \\ \delta_q \end{pmatrix} = \begin{pmatrix} -\mathfrak{F}_1^k \\ -\mathfrak{F}_2^k \\ -\mathfrak{F}_3^k \end{pmatrix} \quad (7.35)$$

## II. Parameter Selection Methods for Total Variation Models

where

$$\begin{aligned} A_k^h &= [D^h(e_N) - D^h(v_k^h)\chi_{\mathcal{A}_{\beta_k}}D^h(\text{sign}(T^h u_k^h - g^h))]T^h, \\ B_k^h &= [D^h(q_k^h)\chi_{\mathcal{A}_{\gamma_k}}D^h(m_{\gamma_k})^{-1}M^h(\nabla^h u_k^h) - D^h((\alpha^h, \alpha^h)^t)]\nabla^h, \\ \mathfrak{F}_1^k &= Tu_k^h - g^h - D^h(m_{\beta_k})v_k^h, \\ \mathfrak{F}_2^k &= -(\nabla^h)^t q_k^h + \kappa \Delta^h u_k^h - \frac{1}{\beta + \mu}(T^h)^t(T^h u_k^h - g^h) - \frac{\mu}{\mu + \beta}(T^h)^t v_k^h, \\ \mathfrak{F}_3^k &= -D^h((\alpha^h, \alpha^h)^t)\nabla^h u_k^h + D^h(m_{\gamma_k})q_k^h, \end{aligned}$$

$e_N \in \mathbb{R}^N$  is the identity vector,  $D^h(v)$  is a diagonal matrix with the vector  $v$  in its diagonal,  $m_{\beta_k} = \max\{\beta, |T^h u_k^h - g^h|\}$ ,  $m_{\gamma_k} = \max\{\gamma \alpha^h, |\nabla^h u_k^h|\}$ ,

$$\begin{aligned} \chi_{\mathcal{A}_{\beta_k}} &= D^h(t_{\beta_k}) \quad \text{with } (t_{\beta_k})_i = \begin{cases} 0 & \text{if } (m_{\beta_k})_i = \beta, \\ 1 & \text{else;} \end{cases} \\ \chi_{\mathcal{A}_{\gamma_k}} &= D^h(t_{\gamma_k}) \quad \text{with } (t_{\gamma_k})_i = \begin{cases} 0 & \text{if } (m_{\gamma_k})_i = \gamma(\alpha^h)_i, \\ 1 & \text{else;} \end{cases} \\ M^h(v) &= \begin{pmatrix} D^h(v_x) & D^h(v_y) \\ D^h(v_x) & D^h(v_y) \end{pmatrix} \quad \text{with } v = (v_x, v_y)^t \in \mathbb{R}^{2N}. \end{aligned}$$

Since the diagonal matrices  $D^h(m_{\beta_k})$  and  $D^h(m_{\gamma_k})$  are invertible, we eliminate  $\delta_v$  and  $\delta_q$  from (7.35), which leads to the following resulting system

$$H_k \delta_u = f_k$$

where

$$\begin{aligned} H_k &:= \frac{1}{\beta + \mu}(T^h)^t T^h - \kappa \Delta^h + \frac{\mu}{\mu + \beta}(T^h)^t D^h(m_{\beta_k})^{-1} A_k^h + (\nabla^h)^t D^h(m_{\gamma_k})^{-1} (-B_k^h), \\ f_k &:= \mathfrak{F}_2^k - \frac{\mu}{\mu + \beta}(T^h)^t D^h(m_{\beta_k})^{-1} \mathfrak{F}_1^k + (\nabla^h)^* D^h(m_{\gamma_k})^{-1} \mathfrak{F}_2^k. \end{aligned}$$

In general  $B_k^h$  and hence  $H_k$  is not symmetric. In [59] it is shown that the matrix  $H_k^h$  at the solution  $(u_k^h, v_k^h, q_k^h) = (\bar{u}, \bar{v}, \bar{q})$  is positive definite whenever

$$[|q_k^h|]_i \leq (\alpha^h)_i \quad \text{and} \quad (|v_k^h|)_i \leq 1 \tag{7.36}$$

for  $i = 1, \dots, N$ .

In case these two inequalities are not satisfied we project  $q_k^h$  and  $v_k^h$  onto their feasible set, i.e.,  $((q_k^h)_i, (q_k^h)_{i+N})$  is set to  $(\alpha^h)_i \max\{(\alpha^h)_i, [|q_k^h|]_i\}^{-1}((q_k^h)_i, (q_k^h)_{i+N})$  and  $(v_k^h)_i$  is replaced by  $\max\{1, (|v_k^h|)_i\}(v_k^h)_i$ . Then the modified system matrix, denoted by  $H_k^+$  is positive definite; see [36]. As pointed out in [58] we may use  $H_k^+ + \varepsilon_k D^h(e_N)$  with  $\kappa = 0$ ,  $\varepsilon_k > 0$  and  $\varepsilon_k \downarrow 0$  as  $k \rightarrow \infty$  instead of  $\kappa > 0$  to obtain a positive definite matrix. Then our semismooth Newton solver may be written as in [58]:

## 7. Automated Parameter Selection for Total Variation Minimization

**semismooth Newton method:** Initialize  $(u_0^h, q_0^h) \in \mathbb{R}^N \times \mathbb{R}^{2N}$  and set  $k := 0$ .

1. Determine the active sets  $\chi_{\mathcal{A}_{\beta_k}} \in \mathbb{R}^{N \times N}$  and  $\chi_{\mathcal{A}_{\gamma_k}} \in \mathbb{R}^{2N \times 2N}$
2. If (7.36) is not satisfied, then compute  $H_k^+$ ; otherwise set  $H_k^+ := H_k$ .
3. Solve  $H_k^+ \delta_u = f_k$  for  $\delta_u$ .
4. Compute  $\delta_q$  by using  $\delta_u$ .
5. Update  $u_{k+1}^h := u_k^h + \delta_u$  and  $q_{k+1}^h := q_k^h + \delta_q$ .
6. Stop or set  $k := k + 1$  and continue with step 1).

This algorithm converges at a superlinear rate, which follows from standard theory; see [52, 59].

In our experiments we always choose  $\kappa = 0$ ,  $\beta = 10^{-3}$ ,  $\gamma = 10^{-2}$ , and  $\mu = 10^6$ .

## 7.6. Numerical Experiments

In the following we present numerical experiments for studying the behavior of the proposed algorithms (i.e., APS-, pAPS-, LATV-, pLATV-algorithm) with respect to its image restoration capabilities and its stability concerning the choice of the initial value  $\alpha_0$ . The performance of these methods is compared quantitatively by means of the peak signal-to-noise-ratio (PSNR) [11], which is widely used as an image quality assessment measure, and the structural similarity measure (MSSIM) [90], which relates to perceived visual quality better than PSNR. When an approximate solution of the  $L^1$ -TV model is calculated, we also compare the restorations by the mean absolute error (MAE), which is an  $L^1$ -based measure defined as

$$\text{MAE} = \|u - \hat{u}\|_{L^1(\Omega)},$$

where  $\hat{u}$  denotes the true image and  $u$  represents the obtained restoration. In general, when comparing PSNR and MSSIM, large values indicate better reconstruction than smaller values, while the smaller MAE becomes the better the reconstruction results are.

Whenever an image is corrupted by Gaussian noise we compute a solution by means of the (multiscale)  $L^2$ -TV model, while for images containing impulsive noise the (multiscale)  $L^1$ -TV model is always considered.

In our numerical experiments the CPS-, APS-, and pAPS-algorithm are terminated as soon as

$$\frac{|\mathcal{H}_\tau(u_{\alpha_n}; g) - \mathcal{B}_\tau(u_{\alpha_n})|}{\mathcal{B}_\tau(u_{\alpha_n})} \leq \epsilon_B = 10^{-5}$$

or the norm of the difference of two successive iterates  $\alpha_n$  and  $\alpha_{n+1}$  drops below the threshold  $\epsilon_\alpha = 10^{-10}$ , i.e.,  $\|\alpha_n - \alpha_{n+1}\| < \epsilon_\alpha$ . The latter stopping criterion is used to terminate the algorithms if  $(\alpha_n)_n$  stagnates and only very little progress is to expect. In fact, if our algorithm converges at least linearly, i.e., if there exists an  $\varepsilon_\alpha \in (0, 1)$  and an  $m > 0$  such that for all  $n \geq m$  we have  $\|\alpha_{n+1} - \alpha_\infty\| < \varepsilon_\alpha \|\alpha_n - \alpha_\infty\|$ , the second stopping criterion at least ensures that the distance between our obtained result  $\alpha$  and  $\alpha_\infty$  is  $\|\alpha - \alpha_\infty\| \leq \frac{\varepsilon_\alpha \|\alpha_m - \alpha_\infty\|}{1 - \varepsilon_\alpha}$ .

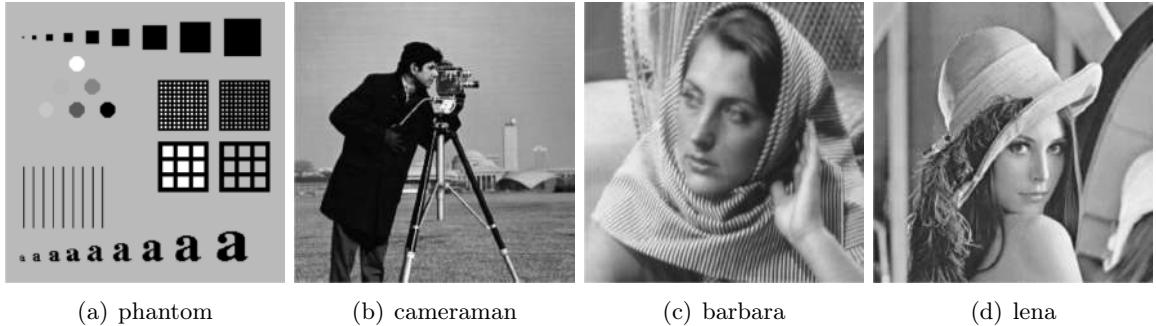
## II. Parameter Selection Methods for Total Variation Models

### 7.6.1. Automatic Scalar Parameter Selection

For automatically selecting the scalar parameter  $\alpha$  in (7.6) we presented in Section 7.3 the APS- and pAPS-algorithm. Here we compare their performance for image denoising and image deblurring.

#### 7.6.1.1. Gaussian Noise Removal

For recovering images corrupted by Gaussian noise with mean zero and standard deviation  $\sigma$  we minimize the functional in (7.6) by setting  $\tau = 2$  and  $T = I$ . Then  $\mathcal{B}_2 = \frac{\sigma^2}{2}|\Omega|$  is a constant independent of  $u$ . The automatic selection of a suitable regularization parameter  $\alpha$  is here performed by the CPS-, APS-, and pAPS-algorithm, where the contained minimization problem is solved by the method presented in Section 7.5.1.1. We recall, that by [18, Theorem 4] and Theorem 7.9 it is ensured that the CPS- and the pAPS-algorithm generate sequences  $(\alpha_n)_n$  which converge to  $\bar{\alpha}$  such that  $u_{\bar{\alpha}}$  solves (7.8). In particular, in the pAPS-algorithm the value  $p$  is chosen in dependency of  $\alpha$ , i.e.,  $p = p(\alpha)$ , such that  $\alpha \rightarrow \frac{(\mathcal{H}_\tau(u_\alpha;g))^{p(\alpha)}}{\alpha}$  is nonincreasing, see Fig. 7.5(a). This property is fundamental for obtaining convergence of this algorithm; see Theorem 7.9. For the APS-algorithm such a monotonic behavior is not guaranteed and hence we cannot ensure its convergence. Nevertheless, if the APS-algorithm generates  $\alpha$ 's such that the function  $\alpha \rightarrow \frac{\mathcal{H}_\tau(u_\alpha;g)}{\alpha}$  is nonincreasing, then it indeed converges to the desired solution, see Theorem 7.11. Unfortunately, the nonincrease of the function  $\alpha \rightarrow \frac{\mathcal{H}_\tau(u_\alpha;g)}{\alpha}$  does not hold always, see Fig. 7.5(b).

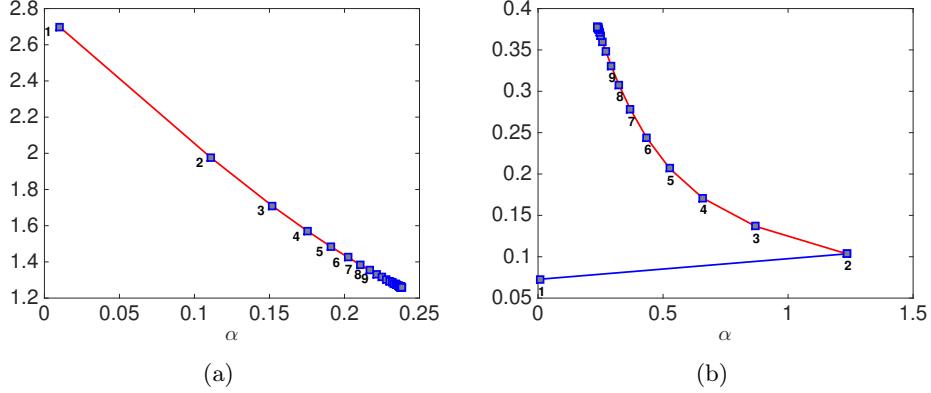


**Figure 7.4:** Original images.

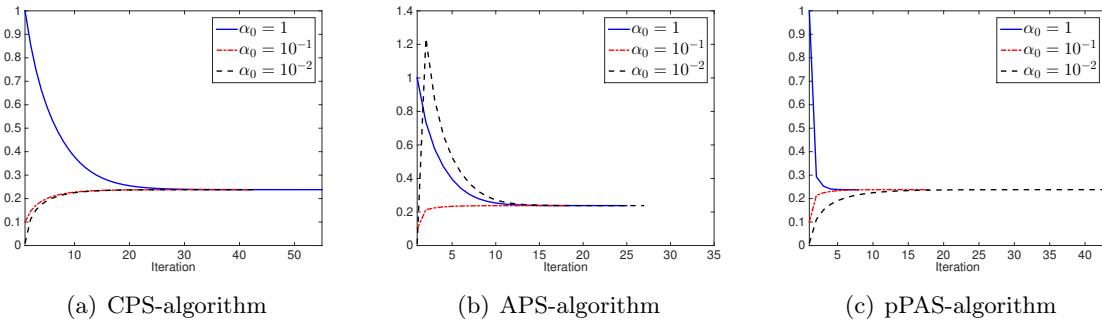
For a performance-comparison here we consider the phantom-image of size  $256 \times 256$  pixels, see Fig. 7.4(a), corrupted only by Gaussian white noise with  $\sigma = 0.3$ . In the pAPS-algorithm we set  $p_0 = 32$ . The behavior of the sequence  $(\alpha_n)_n$  for different initial  $\alpha_0$ , i.e.,  $\alpha_0 \in \{1, 10^{-1}, 10^{-2}\}$  is depicted in Fig. 7.6. We observe that all three methods for arbitrary  $\alpha_0$  converge to the same regularization parameter  $\alpha$  and hence generate results with the same PSNR and MSSIM (i.e., PSNR= 19.84 and MSSIM= 0.7989). Hence, in these experiments, despite the lack of theoretical convergence, also the APS-algorithm seems to converge to the desired solutions. We observe the same behavior for different  $\sigma$  as well.

Looking at the number of iterations needed till termination, we observe from Table 7.1 that the APS-algorithm always needs significantly less iterations than the CPS-algorithm till termination. This is attributed to the different updates of  $\alpha$ . Recall, that for a fixed  $\alpha_n$  in

## 7. Automated Parameter Selection for Total Variation Minimization



**Figure 7.5:** Denoising of the phantom-image corrupted with Gaussian white noise with  $\sigma = 0.03$ . (a) Plot of the function  $\alpha \rightarrow \frac{H_\tau(u_\alpha; g)^p}{\alpha}$  of the pAPS-algorithm with  $\alpha_0 = 10^{-2}$ . (b) Plot of the function  $\alpha \rightarrow \frac{H_\tau(u_\alpha; g)}{\alpha}$  of the APS-algorithm with  $\alpha_0 = 10^{-2}$ . The desired monotone behavior is observed after the second iteration (red part of the curve).



**Figure 7.6:** Denoising of the phantom-image corrupted with Gaussian white noise with  $\sigma = 0.03$ .

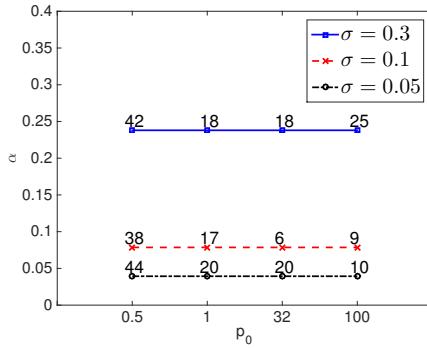
the CPS-algorithm we set  $\alpha_{n+1}^{CPS} := \sqrt{\frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)}} \alpha_n$ , while in the APS-algorithm the update is performed as  $\alpha_{n+1}^{APS} := \frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)} \alpha_n$ . Note, that  $\sqrt{\frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)}} \leq \frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)}$ , if  $\frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)} \geq 1$  and  $\sqrt{\frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)}} \geq \frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)}$ , if  $\frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)} \leq 1$ . Hence, we obtain  $\alpha_n \leq \alpha_{n+1}^{CPS} \leq \alpha_{n+1}^{APS}$  if  $\frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)} \geq 1$  and  $\alpha_n \geq \alpha_{n+1}^{CPS} \geq \alpha_{n+1}^{APS}$  if  $\frac{\nu_2|\Omega|}{\tau H_2(u_{\alpha_n}; g)} \leq 1$ . That is, in the APS-algorithm  $\alpha$  changes more significantly in each iteration than in the CPS-algorithm, which leads to a faster convergence with respect to the number of iterations. Nevertheless, exactly this behavior allows the function  $\alpha \rightarrow \frac{H_2(u_\alpha; g)}{\alpha}$  to increase which is responsible that the convergence of the APS-algorithm is not guaranteed in general. However, in our experiments we observed that the function  $\alpha \rightarrow \frac{H_2(u_\alpha; g)}{\alpha}$  only increases in the first iterations, but nonincreases (actually even decreases) afterwards, see Fig. 7.5(b). This is actually enough to guarantee convergence, as discussed in Section 7.3, since we can consider the solution of the last step in which the desired monotonic behavior is not fulfilled as a “new” initial value. Since from this point on the nonincrease holds, we get convergence of the algorithm.

## II. Parameter Selection Methods for Total Variation Models

$\alpha_0$	$\sigma = 0.3$			$\sigma = 0.1$			$\sigma = 0.05$			$\sigma = 0.01$		
	CPS	APS	pAPS	CPS	APS	pAPS	CPS	APS	pAPS	CPS	APS	pAPS
1	55	25	8	47	21	21	46	20	20	47	17	47
$10^{-1}$	42	18	18	38	17	6	44	20	20	46	18	46
$10^{-2}$	43	27	43	40	20	40	40	18	40	39	17	6
$10^{-3}$	43	31	43	41	22	41	41	19	41	41	20	41
$10^{-4}$	43	35	43	41	23	41	41	22	41	41	19	41

**Table 7.1:** Number of iterations needed for the reconstruction of the phantom-image corrupted by Gaussian white noise with different standard deviations  $\sigma$ . In the pAPS-algorithm we set  $p_0 = 32$ .

The pAPS-algorithm is designed to ensure the nonincrease of the function  $\alpha \rightarrow \frac{(\mathcal{H}_\tau(u_\alpha;g))^{p(\alpha)}}{\alpha}$  by choosing  $p(\alpha)$  in each iteration accordingly, which is done by the algorithm automatically. If  $p(\alpha) = p = 1/2$  in each iteration, then the pAPS-algorithm becomes the CPS-algorithm, as it happens sometimes in practice (indicated by the same number of iterations in Table 7.1). Since the CPS-algorithm converges [18], the pAPS-algorithm always yields  $p \geq 1/2$ . In particular, we observe that if the starting value  $\alpha_0$  is larger than the requested regularization parameter  $\alpha$ , less iteration till termination are needed than with the CPS-algorithm. On the contrary, if  $\alpha_0$  is smaller than the desired  $\alpha$ ,  $p = 1/2$  is chosen by the algorithm to ensure the monotonicity. The obtained result of the pAPS-algorithm is independent on the choice of  $p_0$  as visible from Fig. 7.7. In this plot we also specify the number of iterations needed till termination. On the optimal choice of  $p_0$  with respect to the number of iterations, we conclude from Fig. 7.7 that  $p_0 = 32$  seems to do a good job, although the optimal value may depend on the noise-level.



**Figure 7.7:** Regularization parameter  $\alpha$  obtained by the pAPS-algorithm with different  $p_0$  for denoising the phantom-image corrupted with Gaussian white noise for different  $\sigma$ .

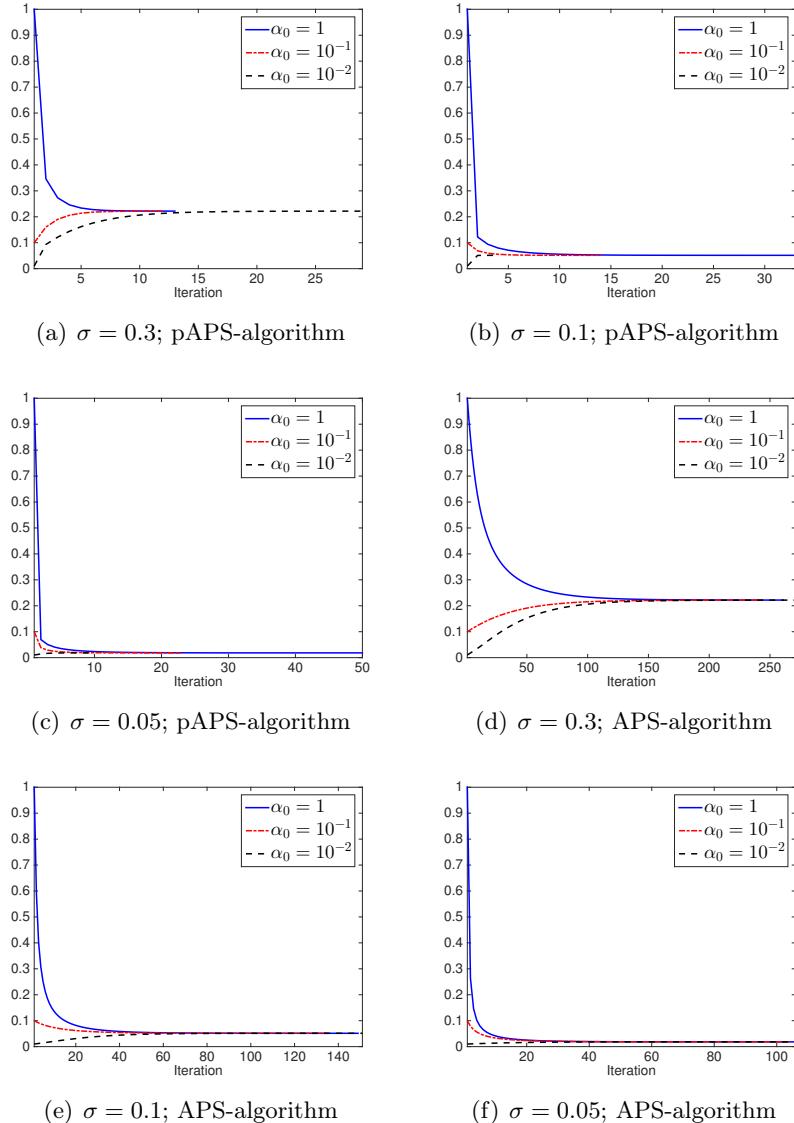
Similar behaviors as described above are also observed for denoising other and real images as well.

### 7.6.1.2. Image Deblurring

Now, we consider the situation when an image is corrupted by some additive Gaussian noise and additionally blurred. Then the operator  $T$  is chosen according to the blurring kernel, which we assume here to be known. For testing the APS- and pAPS-algorithm in this case

## 7. Automated Parameter Selection for Total Variation Minimization

we take the cameraman-image of Fig. 7.4(b), which is of size  $256 \times 256$  pixels, blur it by a Gaussian blurring kernel of size  $5 \times 5$  pixels and standard deviation 10 and additionally add some Gaussian white noise with variance  $\sigma^2$ . The minimization problem in the APS- and pAPS-algorithm is solved approximately by the algorithm in (7.30). In Fig. 7.8 the progress of  $\alpha_n$  for different  $\sigma$ 's, i.e.,  $\sigma \in \{0.3, 0.1, 0.05\}$ , and different  $\alpha_0$ 's, i.e.,  $\alpha_0 \in \{1, 10^{-1}, 10^{-2}\}$  are presented. In these tests both algorithms converge to the same regularization parameter and minimizer. From the figure we observe, that the pAPS-algorithm needs much less iterations than the APS-algorithm till termination. This behavior might be attributed to the choice of the power  $p$  in the pAPS-algorithm, since we observe in all our experiments that  $p > 1$  till termination.



**Figure 7.8:** Reconstruction of the cameraman-image corrupted by Gaussian blurring kernel of size  $5 \times 5$  and standard deviation 10. In the pAPS-algorithm we set  $p_0 = 32$ .

## II. Parameter Selection Methods for Total Variation Models

### 7.6.1.3. Impulsive Noise Removal

It has been demonstrated that for removing impulsive noise in images one should minimize the  $L^1$ -TV model rather than the  $L^2$ -TV model. Then for calculating a suitable regularization parameter  $\alpha$  in the  $L^1$ -TV model we use the APS- and pAPS-algorithm, in which the minimization problems are solved approximately by the  $L^1$ - $TV_\alpha$ -algorithm. Here, we consider the cameraman-image corrupted by salt-and-pepper noise or random-valued impulse noise with different noise-levels, i.e.,  $r_1 = r_2 \in \{0.3, 0.1, 0.05\}$  and  $r \in \{0.3, 0.1, 0.05\}$  respectively. The obtained results for different  $\alpha_0$ 's are depicted in Fig. 7.9 and Fig. 7.10. For the removal of salt-and-pepper noise we observe from Fig. 7.9 similar behaviors of the APS- and pAPS-algorithm as above for removing Gaussian noise. In particular, both algorithms converge to the same regularization parameter. However, in many cases the APS-algorithm needs significantly less iterations than the pAPS-algorithm. These behaviors are also observed in Fig. 7.10 for removing random-valued impulse noise as long as the APS-algorithm finds a solution. In fact, for  $r = 0.05$  it actually does not converge but oscillates as depicted in Fig. 7.10(c).

### 7.6.2. Locally Adaptive Total Variation Minimization

In this section various experiments are presented to evaluate the performance of the LATV- and pLATV-algorithm presented in Section 7.4. Their performance is compared with the proposed pAPS-algorithm as well as with the SA-TV-algorithm introduced in [37] for  $L^2$ -TV minimization and in [58] for  $L^1$ -TV minimization. We recall that the SA-TV methods perform an approximate solution for the optimization problem in (7.10), respectively, and compute automatically a spatially varying  $\lambda$  based on a local variance estimation. However, as pointed out in [37, 58], they only perform efficiently when the initial  $\lambda$  is chosen sufficiently small, as we will do in our numerics. On the contrary, for the LATV- and pLATV-algorithm any positive initial  $\alpha_0$  is sufficient.

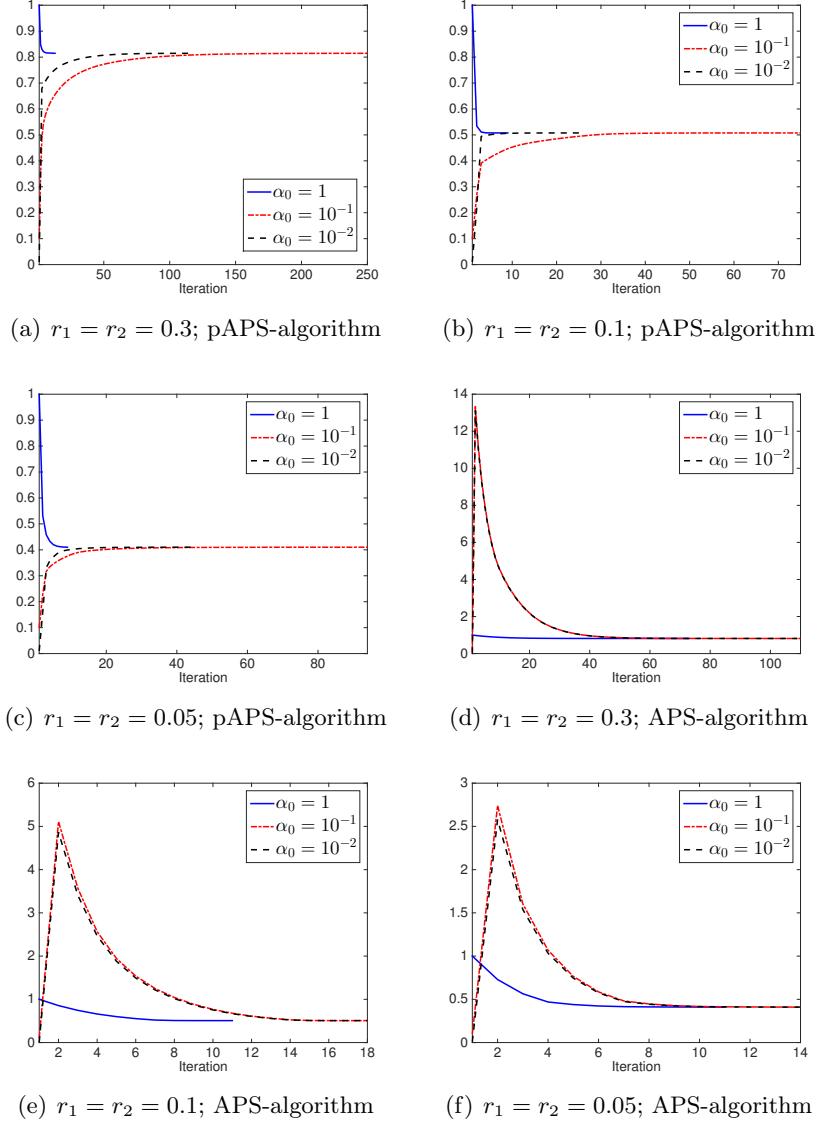
For the comparison we consider four different images, shown in Fig. 7.4, which are all of size  $256 \times 256$  pixels. In all our experiments for the SA-TV-algorithm we use  $\mathcal{I}_{i,j} = \Omega_{i,j}^\omega$ , see [37], and we set the window-size to  $11 \times 11$  pixels in the case of Gaussian noise and to  $21 \times 21$  pixels in case of impulse noise. For the LATV- and pLATV-algorithm we use the window-size  $\omega = 11$ , if not otherwise specified, and choose  $p_0 = \frac{1}{2}$ .

### 7.6.3. Gaussian Noise Removal

#### 7.6.3.1. Dependency on the Initial Regularization Parameter

We start this section by investigating the stability of the SA-TV-, LATV-, and pLATV-algorithm with respect to the initial regularization parameter, i.e.,  $\lambda_0$  for the SA-TV-algorithm and  $\alpha_0$  for the other algorithms, by denoising the cameraman-image corrupted by Gaussian white noise with standard deviation  $\sigma = 0.1$ . In this context we also compare the difference of the pLATV-algorithm with and without using Algorithm 1 for computing automatically an initial parameter, where we set  $c_{\alpha_0} = \frac{1}{5}$ . The minimization problems contained in the LATV- and pLATV-algorithm are solved as described in Section 7.5.1.1. For comparison reasons we define the values  $\text{PSNR}_{\text{diff}} := \max_{\alpha_0} \text{PSNR}(\alpha_0) - \min_{\alpha_0} \text{PSNR}(\alpha_0)$  and  $\text{MSSIM}_{\text{diff}} := \max_{\alpha_0} \text{MSSIM}(\alpha_0) - \min_{\alpha_0} \text{MSSIM}(\alpha_0)$  to measure the variation of the considered quality measures. Here  $\text{PSNR}(\alpha_0)$  and  $\text{MSSIM}(\alpha_0)$  are the PSNR and MSSIM values of the reconstructions, which are obtained from the considered algorithms when the initial regularization parameter

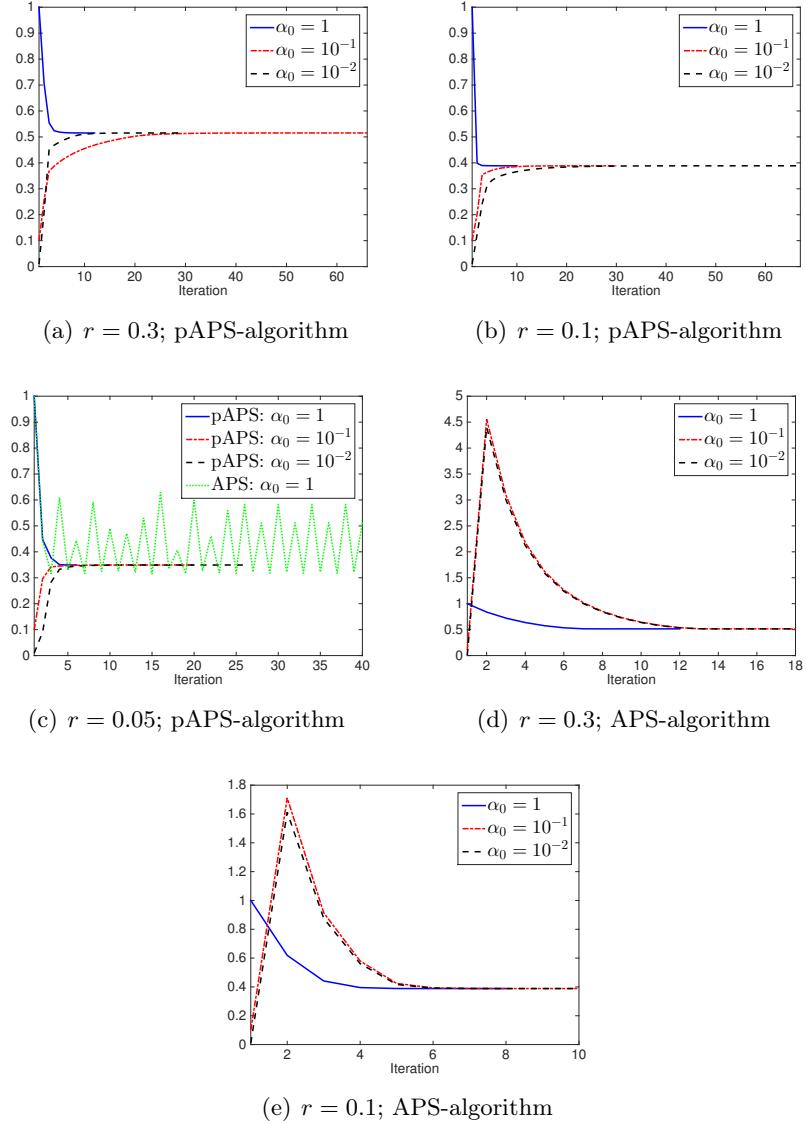
## 7. Automated Parameter Selection for Total Variation Minimization



**Figure 7.9:** Denoising of the cameraman-image corrupted by salt-and-pepper noise. In the pAPS-algorithm we set  $p_0 = 32$ .

is set to  $\alpha_0$ . From Table 7.2 we observe that the pLATV-algorithm with and without Algorithm 1 are more stable with respect to the initial regularization parameter than the LATV-algorithm and the SA-TV-algorithm. This stable performance of the pLATV-algorithm is reasoned by the adaptivity of the value  $p$ , which allows the algorithm to reach the desired residual (at least very closely) for any  $\alpha_0$ . As expected, the pLATV-algorithm with Algorithm 1 is even more stable with respect to  $\alpha_0$  than the pLATV-algorithm alone, since, due to Algorithm 1, the difference of the actually used initial parameters in the pLATV-algorithm is rather small leading to very similar results. Note, that if  $\alpha_0$  is sufficiently small, then the pLATV-algorithm with and without Algorithm 1 coincide, see Table 7.2 for  $\alpha_0 \in \{10^{-2}, 10^{-3}, 10^{-4}\}$ . Actually

## II. Parameter Selection Methods for Total Variation Models



**Figure 7.10:** Denoising of the cameraman-image corrupted by random-valued impulse noise. In the pAPS-algorithm we set  $p_0 = 32$ .

in the rest of our experiments we choose  $\alpha_0$  always so small that Algorithm 1 returns the inputted  $\alpha_0$ .

### 7.6.3.2. Dependency on the Local Window

In Table 7.3 we report on the performance-tests of the pLATV-algorithm with respect to the chosen type of window, i.e.,  $\mathcal{I}_{i,j} = \tilde{\Omega}_{i,j}^\omega$  and  $\mathcal{I}_{i,j} = \Omega_{i,j}^\omega$ . We observe that independently which type of window is used the algorithm finds nearly the same reconstruction. This may be attributed to the fact that the windows in the interior are the same for both types of window. Nevertheless, the boundaries are treated differently, which leads to different theoretical results,

## 7. Automated Parameter Selection for Total Variation Minimization

$\alpha_0/\lambda_0$	SA-TV		LATV		pLATV		pLATV with Algorithm 1	
	PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM
1	27.82	0.8155	27.44	0.8258	27.37	0.8260	27.37	0.8168
$10^{-1}$	27.77	0.8123	27.59	0.8211	27.41	0.8189	27.38	0.8166
$10^{-2}$	27.71	0.8107	27.39	0.8167	27.37	0.8167	27.37	0.8167
$10^{-3}$	27.42	0.8007	27.40	0.8167	27.38	0.8168	27.38	0.8168
$10^{-4}$	27.56	0.7792	27.40	0.8168	27.38	0.8168	27.38	0.8168
PSNR <sub>diff</sub>	0.39646		0.20257		0.044473		0.012704	
MSSIM <sub>diff</sub>	0.036322		0.0091312		0.0092963		0.00019843	

**Table 7.2:** PSNR and MSSIM of the reconstruction of the cameraman-image corrupted by Gaussian white noise with standard deviation  $\sigma = 0.1$  via the LATV- and pLATV-algorithm with different  $\alpha_0$  and via the SA-TV-algorithm with different  $\lambda_0$ . In the LATV- and pLATV-algorithm we use  $\mathcal{I}_{i,j} = \tilde{\Omega}_{i,j}$  with window-size  $11 \times 11$  pixels in the interior and we set  $p_0 = \frac{1}{2}$ .

Image	$\sigma$	pLATV with $\mathcal{I} = \tilde{\Omega}$		pLATV with $\mathcal{I} = \Omega$	
		PSNR	MSSIM	PSNR	MSSIM
cameraman	0.3	22.47	0.6807	22.47	0.6809
	0.1	27.38	0.8168	27.37	0.8165
	0.05	30.91	0.8875	30.92	0.8875
	0.01	40.69	0.9735	40.68	0.9735
lena	0.3	22.31	0.5947	22.30	0.5950
	0.1	26.85	0.7447	26.87	0.7448
	0.05	30.15	0.8301	30.15	0.8300
	0.01	39.69	0.9699	39.68	0.9699

**Table 7.3:** PSNR and MSSIM of the reconstruction of different images corrupted by Gaussian white noise with standard deviation  $\sigma$  via the pLATV-algorithm with  $\alpha_0 = 10^{-4}$ .

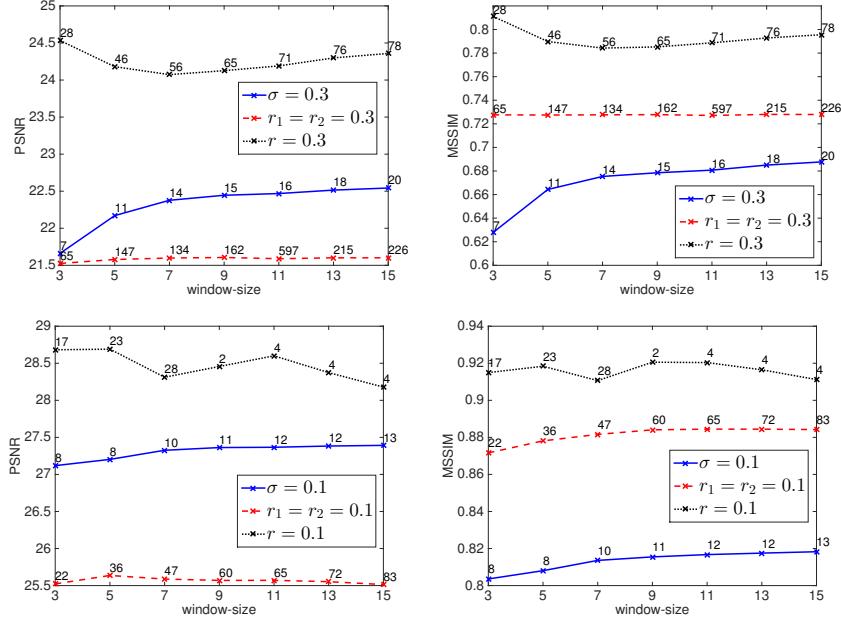
but seems not to have significant influence on the practical behavior. A similar behavior is observed for the LATV-algorithm, as the LATV- and pLATV-algorithm return nearly the same reconstructions as observed below in Table 7.4. Since for both types of windows nearly the same results are obtained, in the rest of our experiments we limit ourselves to always set  $\mathcal{I}_{i,j} = \tilde{\Omega}_{i,j}^\omega$  in the LATV- and pLATV-algorithm.

Next, we test the pLATV-algorithm for different values of the window-size varying from 3 to 15. Fig. 7.11 shows the PSNR and MSSIM of the restoration of the cameraman-image degraded by different types of noise (i.e., Gaussian noise with  $\sigma = 0.3$  or  $\sigma = 0.1$ , salt-and-pepper noise with  $r_1 = r_2 = 0.3$  or  $r_1 = r_2 = 0.1$ , or random-valued impulse noise with  $r = 0.3$  or  $r = 0.1$ ), where the pLATV-algorithm with  $\alpha_0 = 10^{-2}$  and  $p_0 = 1/2$  is used. We observe that the PSNR and MSSIM are varying only slightly with respect to changing window-size. However, in the case of Gaussian noise elimination the PSNR and MSSIM increases very slightly with increasing window-size, while in the case of impulse noise contamination such a behavior cannot be observed. In Fig. 7.11 we also specify the number of iterations needed till termination of the algorithm. From this we observe that a larger window-size results in most experiments in more iterations.

### 7.6.3.3. Homogeneous Noise

Now we test the algorithms for different images corrupted by Gaussian noise with zero mean and different standard deviations  $\sigma$ , i.e.,  $\sigma \in \{0.3, 0.1, 0.05, 0.01\}$ . The initial regularization

## II. Parameter Selection Methods for Total Variation Models



**Figure 7.11:** Restoration of the cameraman-image corrupted by different types of noise via the pLATV-method with different window-sizes.

parameter  $\alpha_0$  is set to  $10^{-4}$  in the pAPS-, LATV-, and pLATV-algorithm. In the SA-TV-algorithm we choose  $\lambda_0 = 10^{-4}$ , which seems sufficiently small. From Table 7.4 we observe that all considered algorithms behave very similar. However, for  $\sigma \in \{0.1, 0.05, 0.01\}$  the SA-TV-algorithm most of the times performs best with respect to PSNR and MSSIM, while sometimes the LATV- and pLATV-algorithm have larger PSNR and MSSIM. That is, looking at these quality measures a locally varying regularization weight is preferred to a scalar one, as long as  $\sigma$  is sufficiently small. In Fig. 7.12 we present the reconstructions obtained via the considered algorithms and we observe that the LATV- and pLATV-algorithm generate visually the best results, while the result of the SA-TV-algorithm seems in some parts over-smoothed. For example, the very left tower in the SA-TV-reconstruction is completely vanished. This object is in the other restorations still visible. For large standard deviations, i.e.  $\sigma = 0.3$ , we observe from Table 7.4 that the SA-TV method performs clearly worse than the other methods, while the pAPS-algorithm usually has larger PSNR and the LATV- and pLATV-algorithm have larger MSSIM. Hence, whenever the noise-level is too large and details are considerably lost due to noise, the locally adaptive methods are not able to improve the restoration quality.

### 7.6.3.4. Nonhomogeneous Noise

For this experiment we consider the cameraman-image degraded by Gaussian white noise with variance  $\sigma^2 = 0.0025$  in the whole domain  $\Omega$  except a rather small area (highlighted in red in Fig. 7.13(a)), denoted by  $\tilde{\Omega}$ , where the variance is 6 times larger, i.e.,  $\sigma^2 = 0.015$  in this part. Since the noise-level is in this application not homogeneous, the pLATV-algorithm presented in Section 7.4 has to be adjusted to this situation accordingly. This can be done by making  $\nu_\tau$  (here  $\tau = 2$ ) locally dependent and we write  $\nu_\tau = \nu_\tau(\hat{u})(x)$  to stress the dependency on the

## 7. Automated Parameter Selection for Total Variation Minimization

Image	$\sigma$	pAPS (scalar $\alpha$ )		SA-TV		LATV		pLATV	
		PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM
phantom	0.3	19.84	0.7989	19.83	0.8319	<b>20.35</b>	0.8411	20.31	<b>0.8432</b>
	0.1	28.97	0.9644	28.97	0.9648	<b>29.50</b>	<b>0.9680</b>	<b>29.50</b>	<b>0.9680</b>
	0.05	34.97	0.9887	33.77	0.9867	<b>35.51</b>	<b>0.9882</b>	<b>35.51</b>	<b>0.9882</b>
	0.01	48.88	<b>0.9994</b>	47.38	0.9987	49.46	0.9993	<b>49.53</b>	0.9993
cameraman	0.3	<b>22.62</b>	<b>0.6911</b>	22.03	0.6806	22.47	0.6807	22.47	0.6807
	0.1	27.31	0.8109	<b>27.56</b>	0.7792	27.40	0.8168	27.38	<b>0.8168</b>
	0.05	30.75	0.8788	<b>31.60</b>	<b>0.8929</b>	30.95	0.8878	30.91	0.8875
	0.01	40.51	0.9731	<b>40.92</b>	0.9649	40.73	<b>0.9737</b>	40.69	0.9735
barbara	0.3	<b>21.22</b>	0.5022	19.78	0.4470	21.05	<b>0.5032</b>	21.05	<b>0.5032</b>
	0.1	24.70	0.7145	<b>25.53</b>	<b>0.7292</b>	24.93	0.7278	24.93	0.7278
	0.05	28.22	0.8514	<b>29.94</b>	<b>0.8801</b>	28.49	0.8584	28.49	0.8584
	0.01	38.91	0.9791	<b>40.56</b>	<b>0.9809</b>	39.08	0.9788	39.08	0.9788
lena	0.3	<b>22.42</b>	0.5930	21.09	0.5474	22.33	<b>0.5951</b>	22.31	0.5947
	0.1	26.84	0.7393	<b>27.31</b>	<b>0.7528</b>	26.85	0.7447	26.85	0.7447
	0.05	30.06	0.8261	<b>30.92</b>	<b>0.8385</b>	30.16	0.8307	30.15	0.8301
	0.01	39.62	0.9685	<b>39.81</b>	0.9660	39.76	<b>0.9708</b>	39.69	0.9699

**Table 7.4:** PSNR- and MSSIM-values of the reconstruction of different images corrupted by Gaussian white noise with standard deviation  $\sigma$  via pAPS-, LATV-, pLATV-algorithm with  $\alpha_0 = 10^{-4}$  and SA-TV-algorithm with  $\lambda_0 = 10^{-4}$ . In the LATV- and pLATV-algorithm we use  $\mathcal{I}_{i,j} = \tilde{\Omega}_{i,j}$  with window-size  $11 \times 11$  pixels in the interior and we set  $p_0 = \frac{1}{2}$ .



**Figure 7.12:** Reconstruction of the cameraman-image corrupted by Gaussian white noise with  $\sigma = 0.1$ .

true image  $\hat{u}$  and on the location  $x \in \Omega$  in the image. In particular, for our experiment we set  $\nu_2 = 0.015$  in  $\tilde{\Omega}$ , while  $\nu_2 = 0.0025$  in  $\Omega \setminus \tilde{\Omega}$ . Since  $\nu_\tau$  is now varying, we also have to adjust the definition of  $\mathcal{B}_\tau$  and  $B_\tau$  to

$$\mathcal{B}_\tau(u) := \int_{\Omega} \nu_\tau(u)(x) \, dx \text{ and } B_\tau(u^h) := \sum_{x \in \Omega^h} \nu_\tau(u^h)(x),$$

respectively for the continuous and discrete setting. Making these adaptations allows us to apply the pLATV-algorithm as well as the pAPS-algorithm to the application of removing nonuniform noise.

The reconstructions obtained by the pAPS-algorithm (with  $p_0 = 32$  and  $\alpha_0 = 10^{-2}$ ) and by the pLATV-algorithm (with  $p_0 = 1/2$  and  $\alpha_0 = 10^{-2}$ ) are shown in Figs. 7.13(b) and 7.13(c) respectively. Due to the adaptive choice of  $\alpha$ , see Fig. 7.13(d) where light colors indicate

## II. Parameter Selection Methods for Total Variation Models

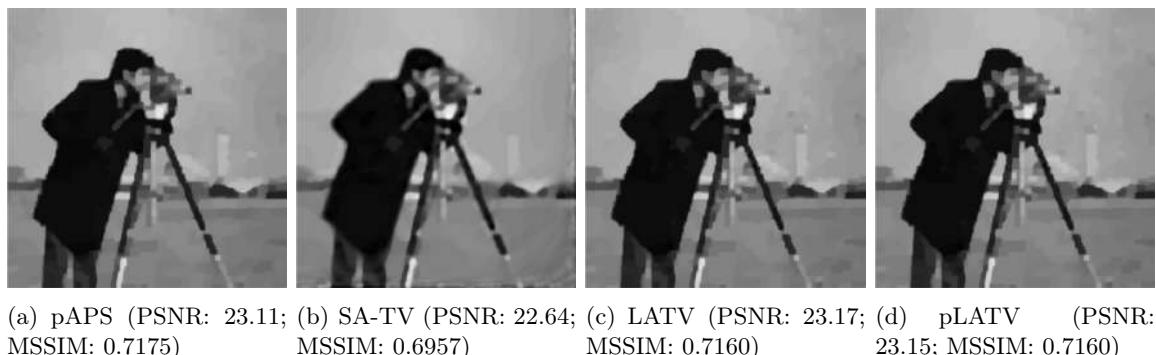
a large value, the pLATV-algorithm is able to remove all the noise considerably, while the pAPS-algorithm returns a restoration, which still retains noise in  $\tilde{\Omega}$ .



**Figure 7.13:** Reconstruction of the cameraman-image corrupted by Gaussian white noise with  $\sigma^2 = 0.0025$  except in the in (a) highlighted area where  $\sigma^2 = 0.015$ .

### 7.6.4. Deblurring and Gaussian noise removal

The performance of the algorithms for restoring images corrupted by Gaussian blur with blurring kernel of size  $5 \times 5$  pixels and standard deviation 10 and additive Gaussian noise with standard deviation  $\sigma$  is reported in Table 7.5. Here we observe that the LATV- as well as the pLATV-algorithm outperform the SA-TV-algorithm for nearly any example. This observation is also clearly visible in Fig. 7.14, where the SA-TV-algorithm produces a still blurred output. The pAPS-algorithm generates very similar reconstructions as the LATV- and pLATV-algorithm, which is also reflected by similar PSNR and MSSIM. Similarly as before, the pAPS-algorithm performs best when  $\sigma = 0.3$ , while for smaller  $\sigma$  the LATV-algorithm has always the best PSNR.



**Figure 7.14:** Reconstruction of the cameraman-image corrupted by Gaussian white noise with  $\sigma = 0.1$  and Gaussian blur.

## 7. Automated Parameter Selection for Total Variation Minimization

Image	$\sigma$	pAPS (scalar $\alpha$ )		SA-TV		LATV		pLATV	
		PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM
phantom	0.3	16.23	0.6958	15.65	0.6632	<b>16.32</b>	0.6995	16.31	<b>0.6997</b>
	0.1	17.86	0.7775	17.31	0.7442	<b>17.98</b>	0.7914	17.97	<b>0.7923</b>
	0.05	18.89	0.7784	19.20	0.8193	<b>19.49</b>	<b>0.8343</b>	19.39	0.8279
cameraman	0.3	<b>21.04</b>	<b>0.6410</b>	19.26	0.5990	20.86	0.6272	20.86	0.6272
	0.1	23.11	<b>0.7175</b>	22.64	0.6957	<b>23.17</b>	0.7157	23.15	0.7156
	0.05	24.14	0.7562	23.75	0.7393	<b>24.22</b>	<b>0.7573</b>	24.21	0.7570
barbara	0.3	<b>20.58</b>	<b>0.4556</b>	18.95	0.4314	20.42	0.4517	20.42	0.4515
	0.1	<b>22.16</b>	0.5589	22.09	<b>0.5687</b>	<b>22.16</b>	0.5597	<b>22.16</b>	0.5597
	0.05	22.87	0.6245	22.88	0.6268	<b>22.92</b>	<b>0.6273</b>	22.90	0.6255
lena	0.3	<b>21.75</b>	<b>0.5542</b>	20.10	0.5278	21.71	0.5529	21.69	0.5528
	0.1	24.44	0.6496	24.39	<b>0.6574</b>	<b>24.50</b>	0.6514	24.49	0.6510
	0.05	25.83	0.7047	25.81	<b>0.7091</b>	<b>25.92</b>	0.7066	25.91	0.7062

**Table 7.5:** PSNR- and MSSIM-values of the reconstruction of different images corrupted by Gaussian blur (blurring kernel of size  $5 \times 5$  pixels with standard deviation 10) and additive Gaussian noise with standard deviation  $\sigma$  via pAPS-, LATV-, pLATV-algorithm with  $\alpha_0 = 10^{-2}$  and SA-TV-algorithm with  $\lambda_0 = 10^{-4}$ . In the LATV- and pLATV-algorithm we use  $\mathcal{I}_{i,j} = \tilde{\Omega}_{i,j}$  with window-size  $11 \times 11$  pixels in the interior and set  $p_0 = \frac{1}{2}$ .

### 7.6.5. Impulse Noise Removal

Since it turns out that the LATV- and pLATV-algorithm produce nearly the same output, here, we compare only our pAPS- and pLATV-algorithm for  $L^1$ -TV minimization, with the SA-TV method introduced in [58], where a semismooth Newton method is used to generate an estimate of the minimizer of (7.10). For the sake of a fair comparison an approximate solution of the minimization problem in the pLATV-algorithm is solved by the semismooth Newton method described in Appendix 7.5.3.1. For the SA-TV method we use the parameters suggested in [58] and hence  $\mathcal{I}_{i,j} = \Omega_{i,j}^\omega$ . Moreover, we set  $\lambda_0 = 0.2$  in our experiments which seems sufficiently small. In Table 7.6 and Table 7.7 we report on the results obtained by the pAPS-, SA-TV-, and pLATV-algorithm for restoring images corrupted by salt-and-pepper noise or random-valued impulse noise, respectively. While the pAPS- and pLATV-algorithm produce quite similar restorations for both type of noises, the SA-TV algorithm seems to be outperformed in most examples. For example, in Fig. 7.15 we observe that the pAPS- and pLATV-algorithm remove the noise considerable while the solution of the SA-TV method still contains noise. On the contrary for the removal of random-valued impulse noise in Fig. 7.16 we see that all three methods produce similar restorations.

## 7.7. Conclusion and Extensions

For  $L^1$ -TV and  $L^2$ -TV minimization including convolution type of problems automatic parameter selection algorithms for scalar and locally dependent weights  $\alpha$  are presented. In particular, we introduce the APS- and pAPS-algorithm for automatically determining a suitable scalar regularization parameter. While for the APS-algorithm its convergence only under some assumptions is shown, the pAPS-algorithm is guaranteed to converge always. Besides the general applicability of these two algorithms they also possess a fast numerical convergence in practice.

In order to treat homogeneous regions differently than fine features in images, which promises a better reconstruction, cf. Proposition 7.15 and Remark 7.16, algorithms for automatically

## II. Parameter Selection Methods for Total Variation Models

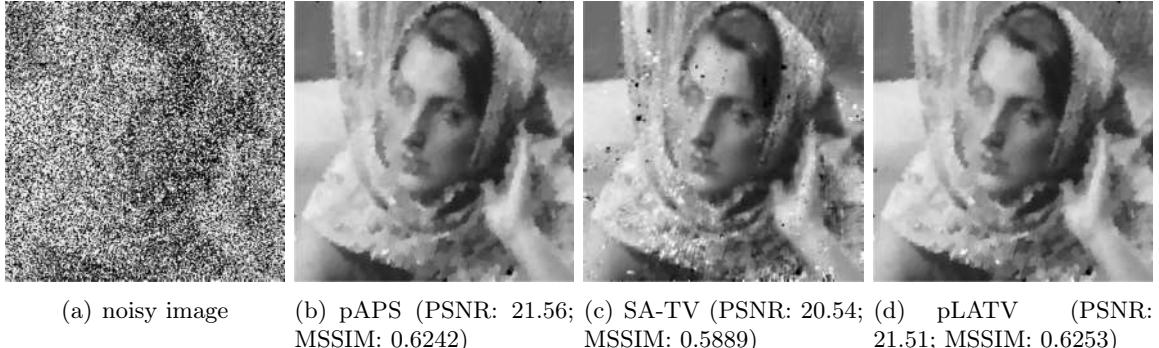
Image	$r_1 = r_2$	pAPS (scalar $\alpha$ )			SA-TV			pLATV		
		PSNR	MSSIM	MAE	PSNR	MSSIM	MAE	PSNR	MSSIM	MAE
phantom	0.3	14.48	0.7040	<b>0.0519</b>	<b>15.28</b>	0.6540	0.0605	<b>14.50</b>	<b>0.7053</b>	<b>0.0519</b>
	0.1	18.39	0.8412	0.0214	<b>19.57</b>	<b>0.8703</b>	<b>0.0196</b>	18.63	0.8610	0.0202
	0.05	21.61	0.9257	<b>0.0103</b>	<b>22.81</b>	<b>0.9362</b>	<b>0.0103</b>	21.55	<b>0.9327</b>	0.0104
cameraman	0.3	<b>21.60</b>	0.7269	<b>0.0343</b>	21.34	0.6871	0.0390	21.59	<b>0.7271</b>	0.0344
	0.1	25.49	0.8822	0.0155	<b>25.80</b>	0.8774	0.0157	25.57	<b>0.8844</b>	<b>0.0154</b>
	0.05	<b>28.80</b>	<b>0.9389</b>	<b>0.0087</b>	28.50	0.9251	0.0095	27.99	0.9263	0.0095
barbara	0.3	<b>21.56</b>	0.6242	<b>0.0486</b>	20.54	0.5889	0.0537	21.51	<b>0.6253</b>	0.0488
	0.1	25.49	0.8729	0.0211	25.27	0.8650	<b>0.0202</b>	<b>25.81</b>	<b>0.8759</b>	0.0208
	0.05	<b>28.46</b>	<b>0.9338</b>	0.0118	27.90	0.9313	<b>0.0110</b>	28.34	0.9325	0.0121
lena	0.3	23.29	0.6807	0.0360	22.61	0.6397	0.0404	<b>23.32</b>	<b>0.6811</b>	<b>0.0359</b>
	0.1	27.60	0.8508	0.0151	27.78	0.8459	0.0152	<b>27.99</b>	<b>0.8530</b>	<b>0.0148</b>
	0.05	29.45	<b>0.8946</b>	<b>0.0096</b>	<b>29.74</b>	0.8863	0.0100	29.53	0.8931	0.0097

**Table 7.6:** PSNR- and MSSIM-values of the reconstruction of different images corrupted by salt-and-pepper noise with  $r_1 = r_2$  via pAPS-, pLATV-algorithm with  $\alpha_0 = 10^{-2}$  and SA-TV-algorithm with  $\lambda_0 = 0.2$  and window-size  $21 \times 21$ . In the pLATV-algorithm we use  $\mathcal{I}_{i,j} = \hat{\Omega}_{i,j}$  with window-size  $11 \times 11$  pixels in the interior and we set  $p_0 = \frac{1}{2}$ .

Image	$r$	pAPS (scalar $\alpha$ )			SA-TV			pLATV		
		PSNR	MSSIM	MAE	PSNR	MSSIM	MAE	PSNR	MSSIM	MAE
phantom	0.3	17.83	0.8120	0.0317	<b>18.68</b>	0.8012	0.0319	18.19	<b>0.8303</b>	<b>0.0305</b>
	0.1	22.46	0.9273	0.0113	<b>23.83</b>	0.9278	<b>0.0100</b>	22.58	<b>0.9328</b>	0.0112
	0.05	25.55	0.9636	0.0058	<b>26.56</b>	0.9642	<b>0.0054</b>	25.45	<b>0.9665</b>	0.0057
cameraman	0.3	<b>24.87</b>	<b>0.8337</b>	<b>0.0213</b>	23.48	0.7583	0.0237	24.19	0.7887	0.0234
	0.1	<b>29.33</b>	<b>0.9359</b>	<b>0.0087</b>	27.72	0.9087	0.0089	28.60	0.9204	0.0093
	0.05	<b>31.46</b>	<b>0.9603</b>	0.0053	30.53	0.9478	<b>0.0052</b>	30.84	0.9442	0.0058
barbara	0.3	<b>24.24</b>	<b>0.8040</b>	0.0301	23.96	0.7977	<b>0.0280</b>	<b>24.24</b>	0.7992	0.0302
	0.1	<b>29.20</b>	<b>0.9355</b>	0.0118	28.60	0.9327	<b>0.0101</b>	28.91	0.9305	0.0120
	0.05	<b>31.95</b>	<b>0.9650</b>	0.0065	30.65	0.9578	<b>0.0059</b>	31.85	0.9640	0.0066
lena	0.3	<b>26.80</b>	<b>0.8124</b>	<b>0.0205</b>	24.74	0.7560	0.0236	26.63	0.8082	0.0208
	0.1	<b>30.34</b>	<b>0.8965</b>	<b>0.0092</b>	28.96	0.8833	0.0095	30.07	0.8918	0.0095
	0.05	<b>31.36</b>	<b>0.9189</b>	0.0062	30.42	0.9180	<b>0.0059</b>	31.08	0.9159	0.0063

**Table 7.7:** PSNR- and MSSIM-values of the reconstruction of different images corrupted by random-valued impulse noise via pAPS-, pLATV-algorithm with  $\alpha_0 = 10^{-2}$  and SA-TV-algorithm with  $\lambda_0 = 0.2$  and window-size  $21 \times 21$ . In the pLATV-algorithm we use  $\mathcal{I}_{i,j} = \hat{\Omega}_{i,j}$  with window-size  $11 \times 11$  pixels in the interior and we set  $p_0 = \frac{1}{2}$  in the pLATV-algorithm.

## 7. Automated Parameter Selection for Total Variation Minimization



**Figure 7.15:** Reconstruction of the barbara-image corrupted by salt-and-pepper noise with  $r_1 = r_2 = 0.3$ .



**Figure 7.16:** Reconstruction of the barbara-image corrupted by random-valued impulse noise with  $r = 0.3$ .

computing locally adapted weights  $\alpha$  are proposed. These methods are much more stable with respect to the initial  $\alpha_0$  than the state-of-the-art SA-TV method. Moreover, while in the SA-TV-algorithm the initial  $\lambda_0 > 0$  has to be chosen sufficiently small, in our proposed methods any arbitrary  $\alpha_0 > 0$  is allowed. Hence the LATV- and pLATV-algorithm are much more flexible with respect to the initialization. By numerical experiments it is shown that the reconstructions obtained by the newly introduced algorithms are similar with respect to image quality measure to the restorations obtained by the SA-TV algorithm. In the case of Gaussian noise removal (including deblurring) for sufficiently small noise-levels reconstructions obtained by locally varying weights seem to be qualitatively better than results with scalar parameters. On the contrary, for removing impulse noise a spatially varying  $\alpha$  or  $\lambda$  is in general not always improving the restoration quality.

For computing a minimizer of the respective multiscale total variation model we present first and second order methods and show their convergence to a respective minimizer.

Although the proposed parameter selection algorithms are constructed to estimate the parameter  $\alpha$  in (7.6) and (7.9), they can be easily adjusted to find a good candidate  $\lambda$  in (7.7) and (7.10), respectively, as well.

## II. Parameter Selection Methods for Total Variation Models

Note, that the proposed parameter selection algorithms are not restricted to total variation minimization, but may be extended to other type of regularizers as well by imposing respective assumptions that guarantee a minimizer of the considered optimization problem. In order to obtain similar (convergence) results as presented in Section 7.3 and Section 7.4 the considered regularizer should be convex, lower semicontinuous and one-homogeneous. In particular for proving convergence results as in Section 7.3 (cf. Theorem 7.9 and Theorem 7.11) an equivalence of the penalized and corresponding constrained minimization problem, as in Theorem 7.3, is needed. An example of such a regularizer for which the presented algorithms are extendable is the total generalized variation [13].

## 7.8. References

- [1] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE Transactions on Signal Processing*, 45(4):913–917, 1997.
- [2] A. Almansa, C. Ballester, V. Caselles, and G. Haro. A TV based restoration model with local constraints. *Journal of Scientific Computing*, 34(3):209–236, 2008.
- [3] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, 2000.
- [4] G. Aubert and J.-F. Aujol. A variational approach to removing multiplicative noise. *SIAM Journal on Applied Mathematics*, 68(4):925–946, 2008.
- [5] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition – modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67(1):111–136, 2006.
- [6] S. D. Babacan, R. Molina, and A. K. Katsaggelos. Parameter estimation in TV image restoration using variational distribution approximation. *IEEE Transactions on Image Processing*, 17(3):326–339, 2008.
- [7] S. Bartels. *Numerical Methods for Nonlinear Partial Differential Equations*, volume 14. Springer, 2015.
- [8] M. Bertalmío, V. Caselles, B. Rougé, and A. Solé. TV based image restoration with local constraints. *Journal of Scientific Computing*, 19(1-3):95–122, 2003.
- [9] P. Blomgren and T. F. Chan. Modular solvers for image restoration problems using the discrepancy principle. *Numerical Linear Algebra with Applications*, 9(5):347–358, 2002.
- [10] T. Blu and F. Luisier. The SURE-LET approach to image denoising. *IEEE Transactions on Image Processing*, 16(11):2778–2786, 2007.
- [11] A. C. Bovik. *Handbook of Image and Video Processing*. Academic press, 2010.
- [12] A. Braides.  *$\Gamma$ -Convergence for Beginners*, volume 22 of *Oxford Lecture Series in Mathematics and its Applications*. Oxford University Press, Oxford, 2002.
- [13] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.
- [14] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.
- [15] J.-F. Cai, R. H. Chan, and M. Nikolova. Two-phase approach for deblurring images corrupted by impulse plus Gaussian noise. *Inverse Problems and Imaging*, 2(2):187–204, 2008.

## 7. Automated Parameter Selection for Total Variation Minimization

- [16] L. Calatroni, C. Chung, J. C. De Los Reyes, C.-B. Schönlieb, and T. Valkonen. Bilevel approaches for learning of variational imaging models. *arXiv preprint arXiv:1505.02120*, 2015.
- [17] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.
- [18] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97, 2004.
- [19] A. Chambolle and J. Darbon. On total variation minimization and surface evolution using parametric maximum flows. *International Journal of Computer Vision*, 84(3):288–307, 2009.
- [20] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, 1997.
- [21] A. Chambolle and T. Pock. On the ergodic convergence rates of a first-order primal–dual algorithm. *Mathematical Programming*, pages 1–35.
- [22] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [23] T. F. Chan and S. Esedoḡlu. Aspects of total variation regularized  $L^1$  function approximation. *SIAM Journal on Applied Mathematics*, 65(5):1817–1837, 2005.
- [24] T. F. Chan, G. H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.
- [25] T. F. Chan, J. Shen, and H.-M. Zhou. Total variation wavelet inpainting. *Journal of Mathematical Imaging and Vision*, 25(1):107–125, 2006.
- [26] V. C. Chung, J. C. De Los Reyes, and C.-B. Schönlieb. Learning optimal spatially-dependent regularization parameters in total variation image restoration. *arXiv preprint arXiv:1603.09155*, 2016.
- [27] P. G. Ciarlet. *Introduction to Numerical Linear Algebra and Optimisation*. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge, 1989.
- [28] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.
- [29] J. Darbon and M. Sigelle. A fast and exact algorithm for total variation minimization. In *Pattern Recognition and Image Analysis*, pages 351–359. Springer, 2005.
- [30] J. Darbon and M. Sigelle. Image restoration with discrete constrained total variation. I. Fast and exact optimization. *Journal of Mathematical Imaging and Vision*, 26(3):261–276, 2006.
- [31] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communication on Pure and Applied Mathematics*, 57(11):1413–1457, 2004.
- [32] I. Daubechies, G. Teschke, and L. Vese. Iteratively solving linear inverse problems under general convex constraints. *Inverse Problems and Imaging*, 1(1):29–46, 2007.
- [33] J. C. De Los Reyes and C.-B. Schönlieb. Image denoising: Learning the noise model via nonsmooth PDE-constrained optimization. *Inverse Problems and Imaging*, 7(4), 2013.
- [34] C.-A. Deledalle, S. Vaiter, J. Fadili, and G. Peyré. Stein unbiased gradient estimator of the risk (sugar) for multiple parameter selection. *SIAM Journal on Imaging Sciences*, 7(4):2448–2487, 2014.
- [35] D. C. Dobson and C. R. Vogel. Convergence of an iterative method for total variation denoising. *SIAM Journal on Numerical Analysis*, 34(5):1779–1791, 1997.

## II. Parameter Selection Methods for Total Variation Models

- [36] Y. Dong, M. Hintermüller, and M. Neri. An efficient primal-dual method for  $L^1$  TV image restoration. *SIAM Journal on Imaging Sciences*, 2(4):1168–1189, 2009.
- [37] Y. Dong, M. Hintermüller, and M. M. Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *Journal of Mathematical Imaging and Vision*, 40(1):82–104, 2011.
- [38] D. L. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224, 1995.
- [39] I. Ekeland and R. Témam. *Convex Analysis and Variational Problems*, volume 28 of *Classics in Applied Mathematics*. SIAM, Philadelphia, PA, english edition, 1999.
- [40] Y. C. Eldar. Generalized sure for exponential families: Applications to regularization. *IEEE Transactions on Signal Processing*, 57(2):471–481, 2009.
- [41] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*, volume 375 of *Mathematics and its Applications*. Kluwer Academic Publishers Group, Dordrecht, 1996.
- [42] M. Fornasier, V. Naumova, and S. V. Pereverzyev. Parameter choice strategies for multipenalty regularization. *SIAM Journal on Numerical Analysis*, 52(4):1770–1794, 2014.
- [43] P. Getreuer, M. Tong, and L. A. Vese. A variational model for the restoration of MR images corrupted by blur and Rician noise. In *Advances in Visual Computing*, pages 686–698. Springer, 2011.
- [44] G. Gilboa, N. Sochen, and Y. Y. Zeevi. Texture preserving variational denoising using an adaptive fidelity term. In *Proceedings VLsM*, volume 3, 2003.
- [45] R. Giryes, M. Elad, and Y. C. Eldar. The projected GSURE for automatic parameter tuning in iterative shrinkage methods. *Applied and Computational Harmonic Analysis*, 30(3):407–422, 2011.
- [46] E. Giusti. *Minimal Surfaces and Functions of Bounded Variation*, volume 80 of *Monographs in Mathematics*. Birkhäuser Verlag, Basel, 1984.
- [47] T. Goldstein and S. Osher. The split Bregman method for  $L_1$ -regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.
- [48] G. H. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.
- [49] P. C. Hansen. Analysis of discrete ill-posed problems by means of the  $L$ -curve. *SIAM Review*, 34(4):561–580, 1992.
- [50] P. C. Hansen and D. P. O’Leary. The use of the  $L$ -curve in the regularization of discrete ill-posed problems. *SIAM Journal on Scientific Computing*, 14(6):1487–1503, 1993.
- [51] C. He, C. Hu, W. Zhang, and B. Shi. A fast adaptive parameter estimation for total variation image restoration. *IEEE Transactions on Image Processing*, 23(12):4954–4967, 2014.
- [52] M. Hintermüller and K. Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM Journal on Applied Mathematics*, 64(4):1311–1333, 2004.
- [53] M. Hintermüller and A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1/L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.
- [54] M. Hintermüller and A. Langer. Adaptive regularization for Parseval frames in image processing. *SFB-Report No. 2014-014*, page 12, 2014.
- [55] M. Hintermüller and A. Langer. Non-overlapping domain decomposition methods for dual total variation based image denoising. *Journal of Scientific Computing*, 62(2):456–481, 2015.

## 7. Automated Parameter Selection for Total Variation Minimization

- [56] M. Hintermüller and C. Rautenberg. Optimal selection of the regularization function in a generalized total variation model. Part I: Modelling and theory. *WIAS preprint 2235*, 2016.
- [57] M. Hintermüller, C. Rautenberg, T. Wu, and A. Langer. Optimal selection of the regularization function in a generalized total variation model. Part II: Algorithm, its analysis and numerical tests. *WIAS preprint 2236*, 2016.
- [58] M. Hintermüller and M. M. Rincon-Camacho. Expected absolute value estimators for a spatially adapted regularization parameter choice rule in  $L^1$ -TV-based image restoration. *Inverse Problems*, 26(8):085005, 30, 2010.
- [59] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.
- [60] S. Kindermann, S. Osher, and P. W. Jones. Deblurring and denoising of images by nonlocal functionals. *Multiscale Modeling & Simulation*, 4(4):1091–1115, 2005.
- [61] K. Kunisch and T. Pock. A bilevel optimization approach for parameter learning in variational models. *SIAM Journal on Imaging Sciences*, 6(2):938–983, 2013.
- [62] A. Langer. *Subspace Correction and Domain Decomposition Methods for Total Variation Minimization*. PhD thesis, Johannes Kepler Universität Linz, Linz, Austria, 2011.
- [63] A. Langer, S. Osher, and C.-B. Schönlieb. Bregmanized domain decomposition for image restoration. *Journal of Scientific Computing*, 54(2-3):549–576, 2013.
- [64] T. Le, R. Chartrand, and T. J. Asaki. A variational approach to reconstructing images corrupted by Poisson noise. *Journal of Mathematical Imaging and Vision*, 27(3):257–263, 2007.
- [65] F. Li, M. K. Ng, and C. Shen. Multiplicative noise removal with spatially varying regularization parameters. *SIAM Journal on Imaging Sciences*, 3(1):1–20, 2010.
- [66] H. Liao, F. Li, and M. K. Ng. Selection of regularization parameter in total variation image restoration. *Journal of the Optical Society of America A*, 26(11):2311–2320, 2009.
- [67] Y. Lin, B. Wohlberg, and H. Guo. UPRE method for total variation parameter selection. *Signal Processing*, 90(8):2546–2551, 2010.
- [68] D. A. Lorenz and T. Pock. An inertial forward-backward algorithm for monotone inclusions. *Journal of Mathematical Imaging and Vision*, 51(2):311–325, 2015.
- [69] C. L. Mallows. Some comments on  $C_P$ . *Technometrics*, 15(4):661–675, 1973.
- [70] V. A. Morozov. *Methods for Solving Incorrectly Posed Problems*. Springer-Verlag, New York, 1984.
- [71] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communication on Pure and Applied Mathematics.*, 42(5):577–685, 1989.
- [72] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152, 2005.
- [73] M. K. Ng, P. Weiss, and X. Yuan. Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods. *SIAM Journal on Scientific Computing*, 32(5):2710–2736, 2010.
- [74] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40(3):965–994, 2002.
- [75] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1-2):99–120, 2004.

## II. Parameter Selection Methods for Total Variation Models

- [76] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.
- [77] K. Papafitsoros and C.-B. Schönlieb. A combined first and second order variational approach for image reconstruction. *Journal of Mathematical Imaging and Vision*, 48(2):308–338, 2014.
- [78] S. Ramani, Z. Liu, J. Rosen, J. Nielsen, and J. A. Fessler. Regularization parameter selection for nonlinear iterative image restoration and MRI reconstruction using GCV and SURE-based methods. *IEEE Transactions on Image Processing*, 21(8):3659–3672, 2012.
- [79] R. T. Rockafellar. *Convex Analysis*. Princeton Mathematical Series, No. 28. Princeton University Press, Princeton, N.J., 1970.
- [80] L. I. Rudin and S. Osher. Total variation based image restoration with free local constraints. In *Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference*, volume 1, pages 31–35. IEEE, 1994.
- [81] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.
- [82] C. M. Stein. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, pages 1135–1151, 1981.
- [83] D. M. Strong, P. Blomgren, and T. F. Chan. Spatially adaptive local-feature-driven total variation minimizing image restoration. In *Optical Science, Engineering and Instrumentation'97*, pages 222–233. International Society for Optics and Photonics, 1997.
- [84] D. M. Strong and T. F. Chan. Spatially and scale adaptive total variation based regularization and anisotropic diffusion in image processing. In *Division in Image Processing, UCLA Math Department CAM Report*. Citeseer, 1996.
- [85] C. Sutour, C.-A. Deledalle, and J.-F. Aujol. Adaptive regularization of the NL-means: Application to image and video denoising. *IEEE Transactions on Image Processing*, 23(8):3506–3521, 2014.
- [86] E. Tadmor, S. Nezzar, and L. Vese. A multiscale image representation using hierarchical  $(BV, L^2)$  decompositions. *Multiscale Modeling & Simulation*, 2(4):554–579, 2004.
- [87] E. Tadmor, S. Nezzar, and L. Vese. Multiscale hierarchical decomposition of images with applications to deblurring, denoising and segmentation. *Communication in Mathematical Sciences*, 6(2):281–307, 2008.
- [88] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-Posed Problems*. Vh Winston, 1977.
- [89] C. R. Vogel. Non-convergence of the  $L$ -curve regularization parameter selection method. *Inverse Problems*, 12(4):535–547, 1996.
- [90] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
- [91] P. Weiss, L. Blanc-Féraud, and G. Aubert. Efficient schemes for total variation minimization under constraints in image processing. *SIAM Journal on Scientific Computing*, 31(3):2047–2080, 2009.
- [92] Y.-W. Wen and R. H. Chan. Parameter selection for total-variation-based image restoration using discrepancy principle. *IEEE Transactions on Image Processing*, 21(4):1770–1781, 2012.
- [93] M. Zhu and T. Chan. An efficient primal-dual hybrid gradient algorithm for total variation image restoration. *UCLA CAM Report*, pages 08–34, 2008.

## 8. Adaptive Regularization for Parseval Frames in Image Processing

**Bibliographic Note:** The content of this chapter is based on the published report [AL6]: M. Hintermüller and A. Langer. Adaptive regularization for Parseval frames in image processing. *SFB-Report No. 2014-014*, 2014.

A modified version of the aforementioned report is published as a peer-reviewed conference paper [AL7]: M. Hintermüller, A. Langer, C. N. Rautenberg, and T. Wu. Adaptive regularization for image reconstruction from subsampled data. In *Proceedings of the International Conference on Imaging, Vision and Learning Based Optimization and PDEs (ILVOPDE)*, pages 3–26, Springer, 2016.

**Summary:** For image data related to Parseval frames and Gaussian white noise, distributed (or locally adaptive) data fidelity weights for preserving image details while still considerably removing noise from homogeneous image regions in image reconstruction are computed automatically. While the underlying variational model is related to the Rudin-Osher-Fatemi (or total variation) model, the determination of the fidelity weights utilizes localized image residuals. An algorithm, which uses hierarchical image decompositions to speed up the iterative parameter adjustment process, is presented and tested numerically for reconstruction from partial Fourier data and for wavelet inpainting, respectively.

### 8.1. Introduction

Image restoration is one of the fundamental tasks in image processing. Its goal is to reconstruct an image from contaminated data  $f$ , which result from a deterioration due to some (linear) transformation operator  $K$  (such as convolution, wavelet or Fourier transform) applied to the original image  $\hat{u}$  and a subsequent infliction of noise  $\eta$ , i.e.,  $f = K\hat{u} + \eta$ . A popular approach to this restoration task rests on variational methods, i.e., the characterization of the reconstructed image  $u$  as the solution of a minimization problem of the type

$$\min_u \mathcal{D}(u; f) + \alpha \mathcal{R}(u), \quad (8.1)$$

where  $\mathcal{D}(\cdot; f)$  represents a data fidelity term,  $\mathcal{R}(\cdot)$  an appropriate filter and  $\alpha > 0$  a regularization parameter, which balances data fidelity and filtering. The choice of  $\mathcal{D}$  is typically dictated by the type of noise contamination, and  $\mathcal{R}$  follows some prior information.

Here, we focus on noise  $\eta$  with zero mean and quadratic deviation from the mean equal to  $\sigma^2 \geq 0$ . Further, we aim at preserving edges in images well. These two pieces of a priori information lead to

$$\mathcal{D}(u; f) := \frac{1}{2} \|Ku - f\|^2, \quad \text{and} \quad \mathcal{R}(u) = |Du|(\Omega),$$

## II. Parameter Selection Methods for Total Variation Models

where the latter represents the total variation of a function  $u$  (see (8.3) below for its definition) and  $\Omega$  denotes the image domain. Typical choices for the norm in  $\mathcal{D}$  are the  $L^2(\Omega)$ -norm or the Euclidean norm on  $\mathbb{R}^N$ , where the latter is, e.g., relevant when  $K$  is a (finite) wavelet or Fourier transform. The resulting model (8.1) is the well-known Rudin-Osher-Fatemi model [25] which has been studied intensively in the literature; see, e.g., [6, 7, 9, 14, 21, 22, 24, 28, 26] as well as the monograph [32] and the many references therein.

The proper choice of  $\alpha$  is delicate. A general guideline is the following one: Large  $\alpha$ , which would typically be favorable for large homogeneous image regions, not only removes noise, but also details in images. Small  $\alpha$ , on the other hand, might be advantageous in regions with image details, but it adversely retains noise in homogeneous image regions. For an automated choice of  $\alpha$  in (8.1) several methods have been devised; see for example [10, 18, 20, 27, 34] and the references therein. We note that instead of considering (8.1) one may equivalently study  $\lambda\mathcal{D}(u; f) + \mathcal{R}(u)$  with  $\lambda = 1/\alpha$ . Based on this view and considering a piecewise constant function  $\lambda$  over the image domain, where the partitioning of the image into pieces is due to a pre-segmentation, in [3] scalars  $\lambda_i$ ,  $i = 1, \dots, \#\text{pieces}$ , for each segment is computed by an augmented Lagrangian type algorithm. While still remaining in a deterministic regime, interestingly [3] uses a distributed (more precisely a piecewise constant) parameter  $\lambda$ .

Later it was noticed that stable choices of  $\lambda$  resp.  $\alpha$  have to incorporate statistical properties of the noise. In this vein, in [2, 15] automated update rules for  $\lambda$  based on statistics of local constraints were proposed. For statistical multiscale methods we refer to [16, 17, 23]. A different approach has been proposed in [29] for image denoising only, where nonlocal means [5] are used to create a nonlocal data fidelity term. While the methods in [2, 15] are highly competitive in practice, the adjustment of  $\lambda$  relies on the output of  $K$  being a deteriorated image again. This, however, limits the applicability of these approaches in situations where  $K$  transfers an image into a different type of range space. Particular examples of such operators are (discrete) wavelet or Fourier transforms. It is therefore the goal of this paper to study the approach of [15] in the context of wavelet and Fourier transform.

The rest of the paper is organized as follows. In section 8.2 we describe the algorithmic approaches to adaptive regularization. In particular, we revisit the method proposed in [15] and discuss its adaptation to wavelet and Fourier transform. Numerical experiments are reported on in section 8.3.

### 8.2. Adaptive Regularization Approach

In this paper we focus on image restoration problems. With the aim of preserving edges in images, Rudin, Osher and Fatemi [25] proposed total variation regularization in image restoration. In this approach the recovery of the image  $\hat{u}$  is based on minimizing the following constrained optimization problem

$$\min_u |Du|(\Omega) \quad \text{subject to (s.t.)} \quad \int_{\Omega} Kudx = \int_{\Omega} f dx, \quad \int_{\Omega} (Ku - f)^2 dx = \sigma^2 |\Omega|, \quad (8.2)$$

where  $\Omega \subset \mathbb{R}^2$  is a simply connected domain with Lipschitz boundary  $\partial\Omega$  and  $|\Omega|$  its volume, which we assume to be 1 in the sequel for simplicity,  $K$  is a linear bounded operator,  $f = Ku + \eta$  the observed data with  $\hat{u}$  the unknown true signal, and  $\eta$  represents white Gaussian noise with zero mean and standard deviation  $\sigma$ . These statistical properties of  $\eta$  motivate the constraints in (8.2); see, e.g., [7]. Moreover,  $|Du|(\Omega)$  denotes the total variation of  $u \in L^1(\Omega)$  in  $\Omega$ , defined

## 8. Adaptive Regularization for Parseval Frames in Image Processing

by

$$|Du|(\Omega) = \sup\left\{\int_{\Omega} u \operatorname{div} \mathbf{p} dx : \mathbf{p} \in C_0^1(\Omega), \|\mathbf{p}\|_{L^\infty(\Omega)} \leq 1\right\}. \quad (8.3)$$

Here and below,  $L^q(\Omega)$ , with  $q \in [1, \infty]$ , denotes the usual Lebesgue space [1], and  $C_0^l(\Omega)$ ,  $l \in \mathbb{N}$ , is the space of  $l$ -times continuously differentiable functions with compact support in  $\Omega$ . Usually (8.2) is addressed via the following unconstrained optimization problem:

$$\min_u |Du|(\Omega) + \frac{\lambda}{2} \int_{\Omega} |Ku - f|^2 dx \quad (8.4)$$

for a given constant  $\lambda > 0$ . In particular, one can show that there exists a constant  $\lambda \geq 0$  such that the constrained problem (8.2) is equivalent to the unconstrained problem (8.4) if  $K$  does not annihilate constant functions; see [7].

### 8.2.1. The SA-TV Approach

In order to preserve image details as well as to remove noise, in [15] a spatially adaptive total variation approach of (8.4) is studied, where  $\lambda$  is based on local image features and  $K$  is a bounded linear operator from  $L^2(\Omega)$  into  $L^2(\Omega)$  with the output of  $K$  being a deteriorated image. For the ease of reference and comparison with our subsequent development, we provide a brief description of the approach in [15]. In fact, instead of the minimization problem (8.4) one considers the corresponding multiscale version

$$\min_u |Du|(\Omega) + \frac{1}{2} \int_{\Omega} \lambda(x) |Ku - f|^2 dx. \quad (8.5)$$

Supposing that the variance  $\sigma^2$  of the noise is known, then the function  $\lambda$  is chosen automatically and adaptively. In particular, for a discrete domain  $\Omega_m$  containing  $m \times m$  pixels,  $m \in \mathbb{N}$ ,  $f, u \in \mathbb{R}^{m^2}$  and  $K \in \mathbb{R}^{m^2 \times m^2}$  the overall restoration algorithm of [15] can be written in the following way, where  $f, u$ , and  $Ku$  are reshaped as  $m \times m$  matrices:

**SA-TV Algorithm:** Choose  $u_0 = 0 \in \mathbb{R}^{m^2}$ ,  $\lambda_0 \in \mathbb{R}_+^{m^2}$ , and set  $\zeta = 2$  and  $n := 0$

- 1) Compute  $v_n = f - Ku_n$  (if  $n = 0$ , we have  $v_n = f$ ) and solve the discrete version of

$$\arg \min_u |Du|(\Omega) + \frac{1}{2} \int_{\Omega} \lambda(x) |Ku - v_n|^2 dx$$

with a discretization of  $\lambda$  equal to  $\lambda_n$ . Let  $\hat{u}_n$  denote the corresponding minimizer.

- 2) Update  $u_{n+1} := u_n + \hat{u}_n$ .
- 3) Update  $\lambda_{n+1}$  based on  $u_{n+1}$  by setting

$$\begin{aligned} (\tilde{\lambda}_n)_{s,t} &:= (\lambda_n)_{s,t} + \rho_n \left( \sqrt{(\tilde{\mathcal{S}}_n^\omega)_{s,t}} - \sigma \right) \\ (\lambda_{n+1})_{i,j} &= \frac{1}{\omega^2} \sum_{(s,t) \in \Omega_{i,j}^\omega} \zeta \min \left( (\tilde{\lambda}_n)_{s,t}, L \right), \end{aligned} \quad (8.6)$$

where  $L$  is a large positive constant to ensure uniform boundedness of  $(\lambda_n)_{n \in \mathbb{N}}$ ; cf. [15].

- 4) Stop or set  $n := n + 1$  and return to step 1).

Here,

$$\Omega_{i,j}^\omega := \{(s+i, t+j) : -\frac{\omega-1}{2} \leq s, t \leq \frac{\omega-1}{2}\}$$

## II. Parameter Selection Methods for Total Variation Models

is a set of pixel-coordinates in a  $\omega$ -by- $\omega$  window centered at  $(i, j)$  (with a symmetric extension at the boundary) with  $\omega \in \mathbb{N}$  being odd and

$$(\tilde{\mathcal{S}}_n^\omega)_{i,j} := \begin{cases} \mathcal{S}_{i,j}^\omega & \text{if } \mathcal{S}_{i,j}^\omega \geq B, \\ \sigma^2 & \text{otherwise,} \end{cases}$$

where  $\mathcal{S}_{i,j}^\omega := \frac{1}{\omega^2} \sum_{(s,t) \in \Omega_{i,j}^\omega} (f_{s,t} - (Ku_{n+1})_{s,t})^2$  and  $B > 0$  is some bound such that  $\mathcal{S}_{i,j}^\omega > B$  implies that some details in the neighborhood of the pixel  $(i, j)$  are left in the residual image. Utilizing distributions of extremal values (Gumbel distribution), in [15] it was found that a good bound is

$$B := \frac{\sigma^2}{\omega^2} (\mathbb{E}(T_{\max}) + \mathfrak{d}(T_{\max}))$$

where  $T_{\max}$  is the maximum value of  $m^2$  observations distributed along the  $\chi^2$ -distribution with  $\omega^2$ -degrees of freedom,  $\mathbb{E}(\cdot)$  is the expected value and  $\mathfrak{d}(\cdot)$  the standard deviation of a random variable.

It is demonstrated in [15], that this algorithm performs well if  $K = I$  or  $K$  being a blurring operator. In particular, the adjustment of  $\lambda$  relies on the output of  $K$  being an image again. Nevertheless, if  $K$  is a Fourier or orthogonal wavelet transform this algorithm does not generate satisfactory results, see Table 8.1 and Table 8.3. Next, we adjust the above described approach for such situations

### 8.2.2. Our Approach

We consider now the case when  $K = S \circ T$ , where  $T : L^2(\Omega) \rightarrow \ell^2(\Lambda)$  is a Fourier or orthogonal wavelet transform and  $S$  is a subsampling operator from the whole index set  $\Lambda$  onto a known index set  $\tilde{\Lambda} \subset \Lambda$ . Note, that images can be represented, for example, by a sparse wavelet expansion [11], i.e., for a given wavelet basis  $\{\psi_i : i \in \Lambda\}$  the image  $u$  can be well approximated by a series expansion with a few nonvanishing coefficients of the form

$$u \approx T^* u_\Lambda = \sum_{i \in \Lambda} u_i \psi_i$$

where  $u_\Lambda = (u_i)_{i \in \Lambda} \in \ell^2(\Lambda)$  and  $T^*$  is the adjoint operator of  $T$  called synthesis operator. Note that  $T^*$  is again bounded and linear.

In our situation the unconstrained optimisation problem in (8.4) reads as follows

$$\min_u |Du|(\Omega) + \frac{\lambda}{2} \|Ku - f\|_{\ell^2(\tilde{\Lambda})}^2. \quad (8.7)$$

Note, that (8.7) is related to the constrained minimization problem

$$\min |Du|(\Omega) \quad \text{s.t.} \quad \|Ku - f\|_{\ell^2(\tilde{\Lambda})}^2 \leq \sigma^2,$$

where  $\sigma > 0$  is the known noise level.

For our purposes we modify the objective in (8.7) in order to handle the presence of the operator  $K$ . Hence, instead of tackling (8.4) directly we introduce a so-called *surrogate functional* [12] for  $a \in L^2(\Omega)$ , which is defined as

$$\begin{aligned} \mathbb{S}(u, a) &:= |Du|(\Omega) + \frac{\lambda}{2} \left( \|Ku - f\|_{\ell^2(\tilde{\Lambda})}^2 + \delta \|u - a\|_{L^2(\Omega)}^2 - \|K(u - a)\|_{\ell^2(\tilde{\Lambda})}^2 \right) \\ &= |Du|(\Omega) + \frac{\lambda \delta}{2} \|u - f_K(a)\|_{L^2(\Omega)}^2 + \phi(a, K, f, \lambda), \end{aligned} \quad (8.8)$$

## 8. Adaptive Regularization for Parseval Frames in Image Processing

with  $f_K(a) := a - \frac{1}{\delta}K^*(Ka - f) \in L^2(\Omega)$ , and where we assume  $\delta > \|K\|^2$ . Here and below, when  $K$  is an operator,  $\|\cdot\|$  denotes the corresponding operator norm. Moreover,  $\phi$  is a function independent of  $u$ . We observe in (8.8) that the variable  $u$  is not longer directly affected by the action of  $K$ . Rather, minimizing  $\mathbb{S}(u, a)$  for fixed  $a$  resembles a typical image denoising problem. In order to approach a solution of (8.4), we consider the following iteration.

**Surrogate Iteration:** Choose  $u^{(0)} \in L^2(\Omega)$  and compute for  $k = 0, 1, 2, \dots$

$$u^{(k+1)} = \arg \min_u |Du|(\Omega) + \frac{\lambda\delta}{2} \int_{\Omega} |u - f_K^{(k)}|^2 dx. \quad (8.9)$$

with  $f_K^{(k)} := f_K(u^{(k)})$ .

It can be shown that the iterative procedure (8.9) generates a sequence  $(u^{(k)})_{k \in \mathbb{N}}$  which converges to a minimizer of (8.7); see [12, 13]. It is well known that the minimization problem in (8.9) is strictly convex and can be efficiently solved by standard algorithms such as the primal-dual first-order algorithm [6], the split Bregman method [19], or the primal-dual semismooth Newton algorithm [22]. We note that the latter algorithm is able to solve the optimization problem in (8.7) directly.

The problem in (8.9) is related to the globally constrained minimization problem

$$\min |Du|(\Omega) \quad \text{s.t.} \quad \int_{\Omega} |u - f_K^{(k)}|^2 dx \leq A, \quad (8.10)$$

where  $A \in \mathbb{R}_+$  is a constant depending on  $\sigma$  and  $K$ , see [7]. In order to enhance image details while preserving homogeneous regions, we formulate as in [15] the following locally constrained optimization problem

$$\min_u |Du|(\Omega) \quad \text{s.t.} \quad \mathcal{S}(u) - A \leq 0 \quad \text{a.e. in } \Omega, \quad (8.11)$$

where  $\mathcal{S}(u)(x) = \int_{\Omega} w(x, y)(u - f_K^{(k)})^2(y) dy$  with  $w$  being some localization filter defined as in [15]. The constraint function in (8.11) for  $u = u^{(k+1)}$  reads

$$\mathcal{S}(u^{(k+1)}) = \int_{\Omega} w(x, y)(u^{(k+1)} - u^{(k)} + \frac{1}{\delta}K^*(Ku^{(k)} - f))^2(y) dy.$$

Given the convergence result for scalar  $\lambda$  alluded to in connection with (8.9), for  $k \rightarrow \infty$  one may expect that the term  $u^{(k+1)} - u^{(k)}$  vanishes. Then it is enough to require that  $\int_{\Omega} w(x, y)(\frac{1}{\delta}K^*(Ku^{(k)} - f))^2(y) dy \leq A$ . This motivates the locally, i.e., point-wise constrained minimization problem

$$\min_u |Du|(\Omega) \quad \text{s.t.} \quad \int_{\Omega} w(\cdot, y)(\frac{1}{\delta}K^*(Ku - f))^2(y) dy \leq A \quad \text{a.e. in } \Omega. \quad (8.12)$$

Next we discuss the choice of  $A$ . We have that  $\eta = K\hat{u} - f$  and thus  $K^*(K\hat{u} - f) = K^*\eta$ . Since  $K$  is a bounded linear operator, there exists a constant  $C > 0$  such that  $\|Ku\|_{\ell^2(\tilde{\Lambda})} \leq C\|u\|_{L^2(\Omega)}$  for any  $u \in L^2(\Omega)$ . With  $\|K\| = \|K^*\|$  it follows that  $\|K^*(Ku - f)\|_{L^2(\Omega)} \leq C\|Ku - f\|_{\ell^2(\tilde{\Lambda})}$ . This property, the limit argument in (8.12) and (8.10) allow to estimate the value  $A$  from above by  $A \leq \frac{C\sigma^2}{\delta^2}$ . Since  $K = S \circ T$ , where  $S$  is a subsampling operator and  $T$  either

## II. Parameter Selection Methods for Total Variation Models

the Fourier-Transform or an analysis operator of an orthonormal wavelet frame, by Parseval's theorem, i.e.,  $\|K^*(Ku - f)\|_{L^2(\Omega)} = \|Ku - f\|_{\ell^2(\tilde{\Lambda})}$ , we have that  $A = \frac{\sigma^2}{\delta^2}$ . Nevertheless, for such operators  $K$  an updating scheme for  $\lambda$  as described in [15] and Section 8.2.1 is in general not possible. However, we propose a related strategy. For this purpose we set  $\Omega := \Omega_m$  and denote  $r_K(u) = f_K(u) - u$  the discrete residual image with  $r_K(u), f_K(u), u, f \in \mathbb{R}^{m^2}$  and  $K \in \mathbb{R}^{m^2 \times m^2}$ . For convenience we reshape the quantities  $r_K(u), f_K(u), u, f$ , and  $Ku$  as  $m \times m$  matrices. Note, that in our setting  $r_K(u)$  corresponds to  $K^*$  applied to  $Ku - f$ . Since we are looking for a solution  $u^*$  such that  $\sum_{(i,j) \in \Omega_m} (r_K(u^*)_{i,j})^2 = A|\Omega_m|$  we expect that if  $\mathcal{S}_{i,j}^{\omega,K} := \frac{1}{\omega^2} \sum_{(s,t) \in \Omega_{i,j}^\omega} (r_K(u)_{s,t})^2 > A$ , where  $\Omega_{i,j}^\omega$  is defined as above, then  $u$  is an over-smoothed restoration and the residual  $r_K(u)$  contains details. Hence, in our approach we choose  $B := A$ .

If now  $\mathcal{S}_{i,j}^{\omega,K} \in [0, B]$ , then the residual should ideally only contain noise, while otherwise we suppose there are image details contained in the residual image in  $\Omega_{i,j}^\omega$ . Hence we define the following modified local variance estimator

$$\tilde{\mathcal{S}}_{i,j}^{\omega,K} := \begin{cases} \mathcal{S}_{i,j}^{\omega,K} & \text{if } \mathcal{S}_{i,j}^{\omega,K} \geq B, \\ \sigma^2 & \text{otherwise.} \end{cases}$$

Analogous to [15] this leads to the following update rule of  $\lambda$ , where we initially choose  $\lambda$  to be a small positive constant: Let  $\lambda_n \in \mathbb{R}^{m^2}$  be given. Then set

$$(\lambda_{n+1})_{i,j} = \frac{1}{\omega^2} \sum_{(s,t) \in \Omega_{i,j}^\omega} \left( (\lambda_n)_{s,t} + \rho_n \left( \sqrt{(\tilde{\mathcal{S}}_n^{\omega,K})_{s,t}} - \sigma \right) \right), \quad (8.13)$$

where  $\rho_n = \frac{\|\lambda_n\|_{\ell^2(\Omega_m)}}{\sigma}$  denotes a positive scaling parameter. Here and below, we denote by  $\|\cdot\|_{\ell^2(\Omega_m)}$  a weighted Euclidean-norm, i.e.,  $\|u\|_{\ell^2(\Omega_m)}^2 := \frac{1}{|\Omega_m|} \|u\|_2^2$  for any  $u \in \mathbb{R}^{m^2}$ .

In order to keep the number of iterations small we combine a hierarchical decomposition method proposed in [30, 31] by Tadmor, Nezzar and Vese with

$$\min_u J_\lambda^{(k)}(u) := |Du|(\Omega) + \frac{\delta}{2} \int_\Omega \lambda(x)(|u - f_K^{(k)}|)(x) dx.$$

This idea utilizes concepts from interpolation theory to represent a noisy image as the sum of “atoms”  $u_{(\ell)}$ , where every  $u_{(\ell)}$  extracts features at a scale finer than the one of the previous  $u_{(\ell-1)}$ . This method acts like an iterative regularization scheme, i.e., up to some iteration index  $\bar{k}$  the method yields improving reconstruction results with a deterioration (due to noise influence and ill-conditioning) beyond  $\bar{k}$ . Let  $\tilde{J}_\lambda^k$  be a discrete version of  $J_\lambda^k$ . Considering dyadic scales with a scalar  $\lambda$ , for example, the hierarchical decomposition operates in our setting as follows:

1. Choose  $0 < \lambda_0 \in \mathbb{R}^{m^2}$ ,  $u_0 = 0 = u_0^{(0)} \in \mathbb{R}^{m^2}$  and iterate for  $k = 0, 1, \dots, \bar{k}_0 \in \mathbb{N}$

$$u_0^{(k+1)} = \arg \min_u \tilde{J}_{\lambda_0}^{(k)}(u) \quad (8.14)$$

with  $f_K^{(k)} = u_0^{(k)} - \frac{1}{\delta} K^*(Ku_0^{(k)} - f)$ . Set  $u_1 := u_0^{\bar{k}_0+1}$ .

## 8. Adaptive Regularization for Parseval Frames in Image Processing

2. For  $\ell = 1, \dots$  set  $\lambda_\ell := 2^\ell \lambda_0$  and  $v_\ell := f - Ku_\ell$ . Then, for  $\hat{u}_\ell^{(0)} := 0$ , compute for  $k = 0, 1, \dots, \mathfrak{K}_{\ell+1} \in \mathbb{N}$

$$\hat{u}_\ell^{(k+1)} = \arg \min_u \tilde{J}_{\lambda_{\ell+1}}^{(k)}(u)$$

where  $f_K(\hat{u}_\ell^{(k)}) = \hat{u}_\ell^{(k)} - \frac{1}{\delta} K^*(K\hat{u}_\ell^{(k)} - v_\ell)$  in  $\tilde{J}_{\lambda_{\ell+1}}^{(k)}$ . Set  $u_{\ell+1} := u_\ell + \hat{u}_\ell^{(\mathfrak{K}_{\ell+1}+1)}$ .

Similar to (8.6) we correspondingly modify the iterative adaption of  $\lambda$  in (8.13) by setting

$$\begin{aligned} (\tilde{\lambda}_n)_{s,t} &:= (\lambda_n)_{s,t} + \rho_n \left( \sqrt{(\tilde{S}_n^{\omega,K})_{s,t}} - \sigma \right) \\ (\lambda_{n+1})_{i,j} &= \frac{1}{\omega^2} \sum_{(s,t) \in \Omega_{i,j}^\omega} \zeta \min \left( (\tilde{\lambda}_n)_{s,t}, L \right), \end{aligned} \quad (8.15)$$

where  $\zeta \geq 1$  and  $L$  is a large positive constant to ensure uniform boundedness of  $(\lambda_n)_{n \in \mathbb{N}}$ . Then the hierarchical spatial adaptive algorithm can be written as follows.

**sSA-TV Algorithm:** Choose  $u_0 = 0 \in \mathbb{R}^{m^2}$ ,  $\lambda_0 \in \mathbb{R}_+^{m^2}$  and set  $\zeta = 2$  and  $n := 0$

- 1) If  $n = 0$  solve (8.14) with  $u_0^{(0)} := u_0$ ; else compute  $v_n = f - Ku_n$ . Set  $\hat{u}_n^{(0)} := 0$  and compute for  $k = 0, 1, 2, \dots, \mathfrak{K}_n \in \mathbb{N}$

$$\hat{u}_n^{(k+1)} = \arg \min_u \tilde{J}_{\lambda_n}^{(k)}$$

with  $f_K(\hat{u}_n^{(k)}) = \hat{u}_n^{(k)} - \frac{1}{\delta} K^*(K\hat{u}_n^{(k)} - v_n)$  in  $\tilde{J}_{\lambda_n}^{(k)}$ . Let  $\hat{u}_n^{(\mathfrak{K}_n+1)}$  denote the corresponding solution.

- 2) Update  $u_{n+1} := u_n + \hat{u}_n^{(\mathfrak{K}_n+1)}$ .
- 3) Compute residual  $r_K(u_{n+1})$ . If  $\|r_K(u_{n+1})\|_{\ell^2(\Omega_m)}^2 > A$  go to step 4); if  $A > \|r_K(u_{n+1})\|_{\ell^2(\Omega_m)}^2 > 0.9A$  stop; if  $0.9A > \|r_K(u_{n+1})\|_{\ell^2(\Omega_m)}^2$  set  $u_{n+1} := u_n$ ,  $\lambda_n := \lambda_{n-1}$  (unless  $n = 0$ ), and  $\zeta = \max\{0.85\zeta, 1\}$  and continue with step 4)
- 4) Update  $\lambda_{n+1}$  based on  $u_{n+1}$  and (8.15).
- 5) Set  $n := n + 1$  and return to step 1).

Note, that we expect noise in the reconstruction  $u_n$  if the residual  $\|u_n - f_K(u_n)\|_{\ell^2(\Omega_m)}^2$  is much smaller than  $A$ . Therefore we introduce the lower bound  $0.9A$ , i.e., 90 percent of the value  $A$ , which should ideally bound the residual from below. Hence, the sSA-TV algorithm is terminated as soon as the residual drops into the interval  $[0.9A, A]$ . If the squared norm of the residual  $r_K(u_n)$  drops below the lower bound for the first time, we reset  $\zeta = 1.7$  in order to obtain a residual in the next iteration such that  $\|r_K(u_n)\|_{\ell^2(\Omega_m)}^2 \leq \|r_K(u_{n+1})\|_{\ell^2(\Omega_m)}^2$ . In our experiments we even observed that in fact  $\|r_K(u_{n+1})\|_{\ell^2(\Omega_m)}^2 \in [0.9A, A]$ . For solving the minimization problems in the proposed algorithm we use the primal-dual Newton method suggested in [22]. The parameters in the primal-dual Newton algorithm are chosen as  $\mu = 10^{-6}$ ,  $\gamma = 10^{-4}$  and it is terminated as soon as its residual is smaller than  $10^{-4}$ , see [22] for a detailed explanation of the algorithm. In order to obtain a sufficiently good approximation of the minimizer in each iteration  $n$  in step 2), we set  $\mathfrak{K}_n$  adaptively such that  $\|\hat{u}_n^{(\mathfrak{K}_n+1)} - \hat{u}_n^{(\mathfrak{K}_n)}\|_{\ell^2(\Omega_m)} \leq 10^{-6}$ .

## II. Parameter Selection Methods for Total Variation Models

### 8.3. Numerical Experiment

In the following we present numerical experiments for studying the behavior of the sSA-TV algorithm with respect to its image restoration capabilities and its stability concerning the choice of  $\lambda_0$  and the window-size  $\omega$ . We also compare the results obtained by the sSA-TV algorithm with the ones due to the primal-dual Newton algorithm for the experimentally best scalar choice of  $\lambda$  and with the restorations generated by the SA-TV algorithm. The primal-dual Newton algorithm is terminated as soon as the norm of the residual is smaller than  $10^{-4}$  and its parameters are chosen as before, i.e.,  $\mu = 10^{-6}, \gamma = 10^{-4}$ . The performance of these methods is compared quantitatively by means of the peak signal-to-noise- ratio (PSNR) [4], which is widely used as an image quality assessment measure, and the structural similarity measure (MSSIM) [33], which relates to perceived visual quality better than PSNR. In general, when comparing PSNR- and MSSIM- values, large values indicate better reconstruction than smaller values. We also emphasize that for scalar  $\lambda$  other solvers (than primal-dual Newton) may be utilized without changing the conclusions of our findings. This is due to the fact that all of these methods aim at solving problem (8.4).

In our experiments we concentrate on two application: (i) The reconstruction of partial Fourier-data which is a medical image processing task, typically related to magnetic resonance imaging (MRI); (ii) Filling-in of missing or damaged wavelet coefficients, which is sometimes also called *wavelet inpainting* [8, 35].

In all examples in this paper the image intensity range is scaled to  $[0, 1]$ . For both applications we have that  $\|K\| = 1$ , and since  $\delta$  has influence on the convergence speed to the minimizer, i.e., the larger  $\delta$  the slower is the convergence of the surrogate iteration, we set  $\delta = 1.1$  in the sSA-TV Algorithm.

#### 8.3.1. Reconstruction of Partial Fourier-Data

In magnetic resonance imaging one wishes to reconstruct an image which is only given by partial Fourier data and additionally distorted by some additive Gaussian noise with zero mean and standard deviation  $\sigma$ .<sup>1</sup> Hence, the linear bounded operator is  $K = S \circ \mathcal{F}$ , where  $\mathcal{F}$  is the 2D Fourier matrix and  $S$  is a downsampling operator which selects only a few output frequencies.

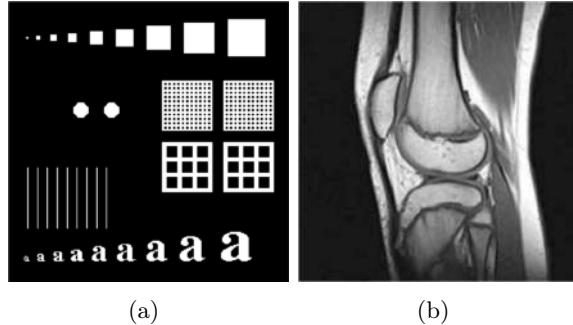
The frequencies are usually sampled along radial lines in the frequency domain, in particular in our experiments along 44, 88, 132, and 176 radial lines, as visualized in Figure 8.2, and some additive Gaussian white noise with zero mean and standard deviation  $\sigma$  is added.

In our first example we consider a phantom-image of size  $256 \times 256$  pixels, see Figure 8.1(a), transformed to its Fourier frequencies. As mentioned before, we sample the frequencies along radial lines and add some additive Gaussian noise with zero mean and standard deviation  $\sigma$ . In particular, we reconstruct the image via the sSA-TV and the SA-TV algorithm by setting  $\lambda_0 = 0.2$  and  $\omega = 11$  and test its performance for different noise-levels, i.e.,  $\sigma \in \{0.3, 0.1, 0.05\}$ . The obtained results are compared with the ones of the primal-dual Newton method with a scalar  $\lambda$ , see Table 8.1. For the latter method we perform many experiments with different scalar  $\lambda$ -values and we present here the results with the largest PSNR- and MSSIM-values, which are in general not achieved by the same scalar  $\lambda$ , i.e., the  $\lambda$ , which yields the largest

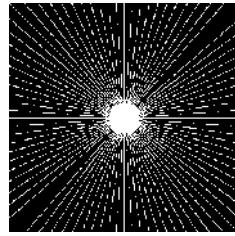
---

<sup>1</sup>Here we assume that  $\|Ku - f\|_{\ell^2(\tilde{\Lambda})}^2$  allows a good approximation of  $\sigma^2$ . Otherwise one has to operate with suitable weighted norms, i.e.,  $\|Ku - f\|_{\ell^2(\tilde{\Lambda})}^2 = \frac{1}{|\tilde{\Lambda}|} \|Ku - f\|_2^2$  with  $|\tilde{\Lambda}|$  small.

## 8. Adaptive Regularization for Parseval Frames in Image Processing



**Figure 8.1:** Images obtained by full Fourier-data (a) “phantom” (b) “knee”



**Figure 8.2:** Sampling domain in the frequency plane, i.e., sampling operator  $S$ .

PSNR, need not necessarily also yield the largest MSSIM. The best values of these two quality measures are summarized in Table 8.1. Note that the SA-TV algorithm gives the worst results with respect to PSNR and MSSIM, while the sSA-TV method typically outperforms the primal-dual Newton method as well as the SA-TV method with respect to these two quality measures. Only when little information is available (strong subsampling) for the reconstruction, then the sSA-TV method may not outperform the best experimental scalar  $\lambda$ . Note that a sampling along 44, 88, 132, and 176 radial lines means that approximately 21%, 39%, 55%, and 69% of the frequency data are available, respectively. Of course, if only very little data is available most details are lost, which explains why our algorithm may not help to improve the reconstruction. However, when we look at the reconstruction for  $\sigma = 0.3$  and a sampling along 132, 88, and 44 radial lines, we see in Figure 8.3 that the sSA-TV improves the restoration in all cases. In particular, in Figure 8.3 we plot the results of the sSA-TV and the “best” results, in the sense of PSNR and MSSIM, of the primal-dual Newton method. The “best” PSNR-result of the primal dual Newton method seems to regularize only a little, which preserves features and therefore gives a high PSNR value, but the restoration contains more noise than the one generated by the sSA-TV. On the other hand the “best” MSSIM-result over-smooths the reconstruction and hence details, like the dots in the squares, are lost in the solution. From this example we observe that the sSA-TV generates a reconstruction which removes noise in particular in the uniform parts and leaves details in the reconstruction at the same time. This observation somehow shows that one may not rely on only one of these quality measures but may consider both or a combination of both as a trustful reference, i.e.,

## II. Parameter Selection Methods for Total Variation Models

$\sigma$	# rays	primal dual Newton		SA-TV method		sSA-TV method	
		PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM
0.3	176	21.19 ( $\lambda = 7$ )	0.710 ( $\lambda = 5$ )	14.79	0.579	<b>22.55</b>	<b>0.865</b>
0.3	132	19.81 ( $\lambda = 10$ )	0.679 ( $\lambda = 5$ )	13.60	0.530	<b>21.24</b>	<b>0.727</b>
0.3	88	<b>18.37</b> ( $\lambda = 12$ )	0.650 ( $\lambda = 5$ )	12.89	0.517	18.19	<b>0.706</b>
0.3	44	16.62 ( $\lambda = 15$ )	<b>0.607</b> ( $\lambda = 5$ )	11.64	0.519	<b>16.85</b>	0.571
0.1	176	29.84 ( $\lambda = 25$ )	0.943 ( $\lambda = 12$ )	23.72	0.882	<b>30.64</b>	<b>0.960</b>
0.1	132	<b>28.32</b> ( $\lambda = 30$ )	0.932 ( $\lambda = 12$ )	21.01	0.706	28.19	<b>0.960</b>
0.1	88	26.22 ( $\lambda = 40$ )	0.905 ( $\lambda = 15$ )	19.10	0.680	<b>26.75</b>	<b>0.941</b>
0.1	44	<b>20.81</b> ( $\lambda = 40$ )	<b>0.775</b> ( $\lambda = 15$ )	16.78	0.556	20.40	0.721
0.05	176	35.61 ( $\lambda = 50$ )	0.981 ( $\lambda = 25$ )	29.93	0.883	<b>35.80</b>	<b>0.983</b>
0.05	132	33.97 ( $\lambda = 50$ )	0.979 ( $\lambda = 25$ )	24.06	0.721	<b>34.92</b>	<b>0.980</b>
0.05	88	<b>31.80</b> ( $\lambda = 75$ )	0.970 ( $\lambda = 25$ )	23.46	0.812	31.12	<b>0.972</b>
0.05	44	22.56 ( $\lambda = 50$ )	<b>0.863</b> ( $\lambda = 30$ )	18.33	0.517	<b>23.39</b>	0.764

**Table 8.1:** For the “phantom”-image we test for different standard deviations  $\sigma$  and different ratios of known Fourier coefficients the solution of the total variation minimization with a constant regularization parameter and compare it with the solution of the adaptive total variation approaches (SA-TV and sSA-TV) with  $\lambda_0 = 0.2$  and  $\omega = 7$  in the sense of the quality measures PSNR and MSSIM.

a reconstruction should have large values in both measures.

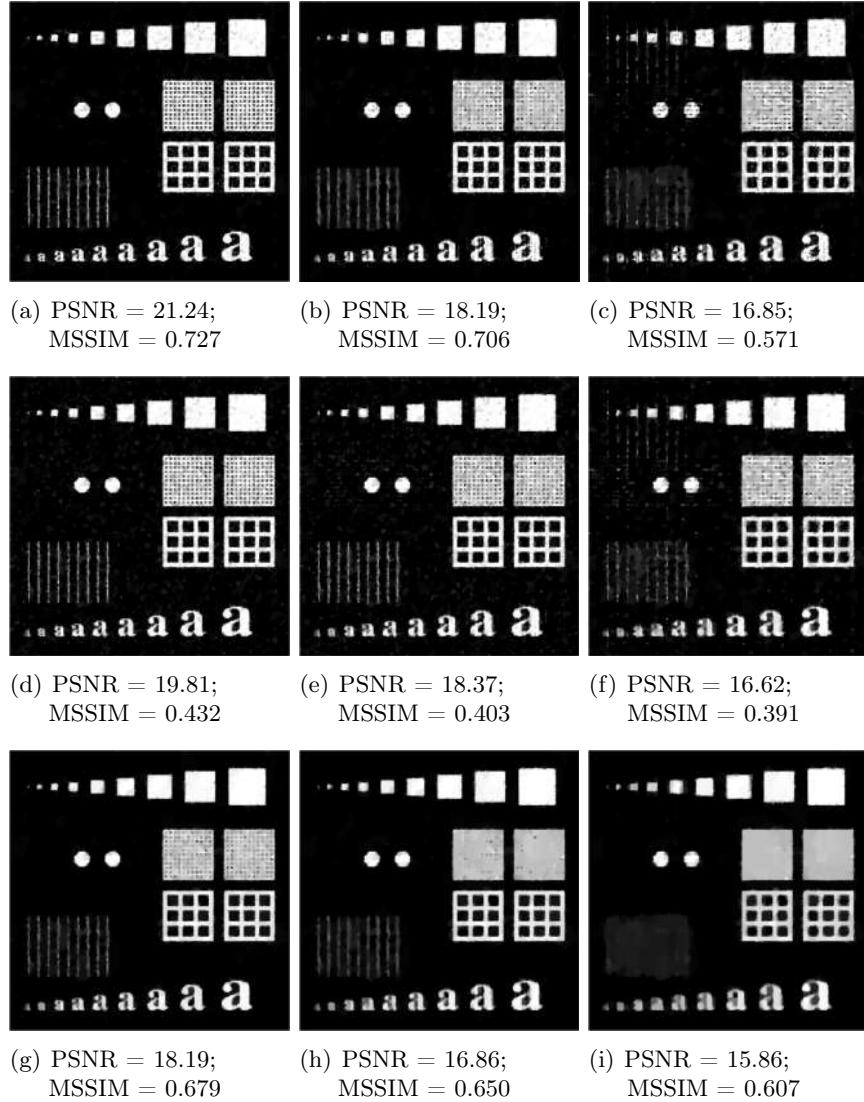
Next we consider an MRI-image of size  $200 \times 200$  pixels of a human knee, see Figure 8.1(b). Similarly as above, the sSA-TV method outperforms the primal-dual Newton method with respect to PSNR and MSSIM when the number of samples is large enough, i.e., when a reasonable amount of frequencies are considered to reconstruct the image, see Table 8.2. Now a sampling along 44, 88, 132, and 176 radial lines means that approximately 26%, 48%, 66%, and 81% of the frequency data are available, respectively. In Figure 8.4 we show the reconstructions of the two methods for a sampling along 88 radial lines and with additive Gaussian noise with zero mean and  $\sigma = 0.1$ . We observe that the sSA-TV method best removes noise in uniform parts while preserving the edges in the image.

Independently of the noise level and the number of radial sampling lines, the final adaptive parameter  $\lambda$  obtained by our parameter choice rule is indeed selected according to the distribution of features in the image. In Figure 8.5 we show the final  $\lambda$ , where light gray indicates large values and dark gray refers to small values. We see that in detail regions  $\lambda$  is large in order to preserve details, while in uniform parts it is small to remove noise considerably.

### 8.3.1.1. Dependence on the Initial Choice $\lambda_0$

Next we investigate our algorithm concerning its stability with respect to the initial  $\lambda_0$ . Therefore we test the sSA-TV method for  $\lambda_0 \in \{0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 1, 1.5, 2, 2.5, 3\}$  and plot the PSNR- and MSSIM-values of the obtained reconstructions in Figure 8.6. Since the parameter  $\lambda$  controls the trade-off between a good data fit and the regularization coming from the TV-term, it has an effect on the variance of the residual  $\|u^{(k)} - z(u^{(k)})\|_{\ell^2(\Omega_m)}$ . This can be seen from the plots in Figure 8.6 where we also specify the number of updates on  $\lambda$  for each  $\lambda_0$ . From our tests we conclude that the sSA-TV method generates reconstructions with large PSNR and MSSIM when the initial  $\lambda_0$  is sufficiently small, while for large  $\lambda_0$  there is

## 8. Adaptive Regularization for Parseval Frames in Image Processing



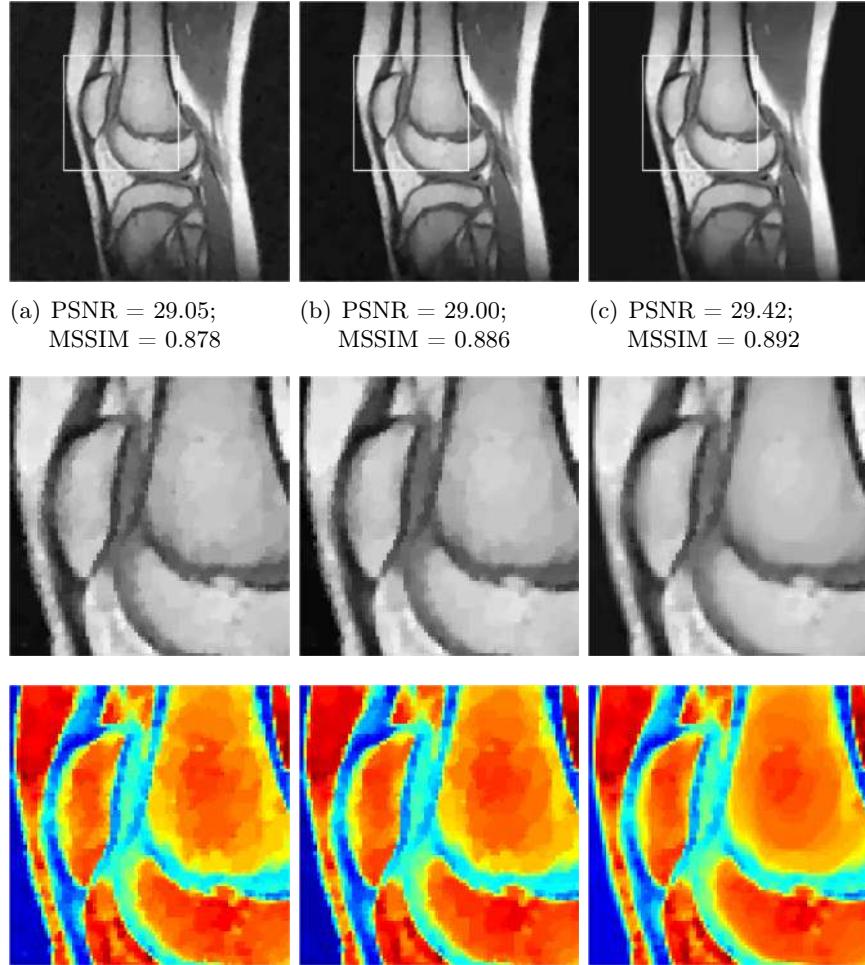
**Figure 8.3:** Results obtained by the sSA-TV (first row) and primal-dual Newton (second row - best PSNR-results; third row - best MSSIM-results) for  $\sigma = 0.3$  and 132 (left column), 88 (mid column), and 44 (right column) radial lines.

no guarantee for a good restoration. This can be attributed to the fact that our  $\lambda$ -update operates by increasing  $\lambda$  from an initial guess. Thus, in case the latter is already too large for producing good reconstructions, the method could only recover if reductions in  $\lambda$  would be allowed as well.

### 8.3.1.2. Dependence on Window Size $\omega$

Next we test our algorithm for different values of the window size  $\omega$  varying from 3 to 23. Figure 8.7 shows the PSNR- and MSSIM- values of the restoration of the phantom image (degraded by Gaussian noise with  $\sigma = 0.1$ ) via the sSA-TV method with  $\lambda_0 = 0.2$ . We observe

## II. Parameter Selection Methods for Total Variation Models



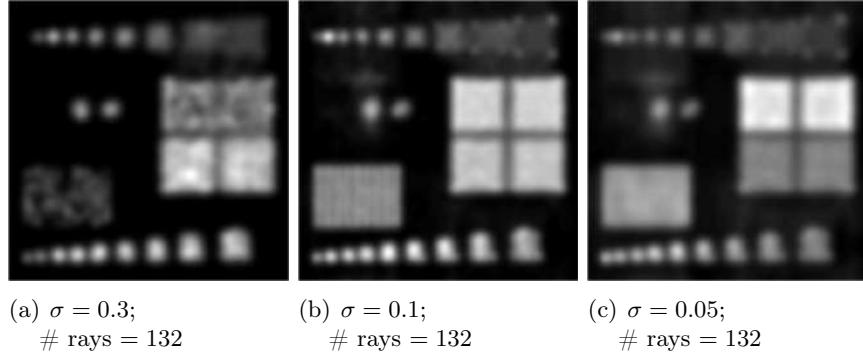
**Figure 8.4:** Reconstruction of the “knee”-image sampled in the Fourier domain along 88 radial lines (corresponds to 48% of the data) which is distorted by Gaussian noise with zero mean and standard deviation  $\sigma = 0.1$ . We show the results of the primal dual TV method with the best PSNR (left column), and the best MSSIM (mid column) and the result of the sSA-TV method (right column). In the second row we zoomed in on the in the first row highlighted area. In order to visualize the differences in the reconstruction we color in the third row the zoomed area.

that the PSNR and MSSIM are varying only slightly with respect to changing window-size. In particular, we observe that whenever the algorithm needs one or more additional steps to terminate, the PSNR exhibits an upwards jump, while the MSSIM seems to behave exactly in the opposite way. This behavior may be attributed to the observation that when the final residual is close to the lower bound  $0.9A$ , larger PSNR and lower MSSIM values are expected than when the residual is close to the upper bound  $A$ . Moreover, large window-sizes, i.e.,  $\omega > 11$ , reduce the MSSIM considerably. These tests recommend to use a rather small window-size as done in our previous experiments, where we set  $\omega = 7$ .

## 8. Adaptive Regularization for Parseval Frames in Image Processing

$\sigma$	# rays	primal dual Newton		sSA-TV method	
		PSNR	MSSIM	PSNR	MSSIM
0.3	176	24.46 ( $\lambda = 5$ )	0.780 ( $\lambda = 5$ )	<b>25.21</b>	<b>0.802</b>
0.3	132	24.44 ( $\lambda = 5$ )	0.785 ( $\lambda = 5$ )	<b>25.07</b>	<b>0.804</b>
0.3	88	24.40 ( $\lambda = 7$ )	0.782 ( $\lambda = 5$ )	<b>24.69</b>	<b>0.793</b>
0.3	44	<b>23.83</b> ( $\lambda = 7$ )	0.759 ( $\lambda = 5$ )	23.51	<b>0.762</b>
0.1	176	29.81 ( $\lambda = 20$ )	0.896 ( $\lambda = 15$ )	<b>30.47</b>	<b>0.909</b>
0.1	132	29.64 ( $\lambda = 20$ )	0.894 ( $\lambda = 20$ )	<b>30.32</b>	<b>0.908</b>
0.1	88	29.05 ( $\lambda = 25$ )	0.886 ( $\lambda = 20$ )	<b>29.42</b>	<b>0.892</b>
0.1	44	27.06 ( $\lambda = 30$ )	0.845 ( $\lambda = 25$ )	<b>27.25</b>	<b>0.853</b>
0.05	176	33.54 ( $\lambda = 20$ )	0.942 ( $\lambda = 40$ )	<b>33.89</b>	<b>0.944</b>
0.05	132	33.08 ( $\lambda = 50$ )	<b>0.940</b> ( $\lambda = 40$ )	<b>33.21</b>	0.938
0.05	88	31.80 ( $\lambda = 50$ )	0.927 ( $\lambda = 50$ )	<b>32.49</b>	<b>0.935</b>
0.05	44	<b>28.71</b> ( $\lambda = 75$ )	0.879 ( $\lambda = 50$ )	28.70	<b>0.883</b>

**Table 8.2:** For the “knee”-image we test for different standard deviations  $\sigma$  and different ratios of known fourier coefficients the solution of the total variation minimization with a constant regularization parameter and compare it with the solution of the adaptive total variation approach with  $\lambda_0 = 0.3$  and  $\omega = 7$  in the sense of the quality measures PSNR and MSSIM.



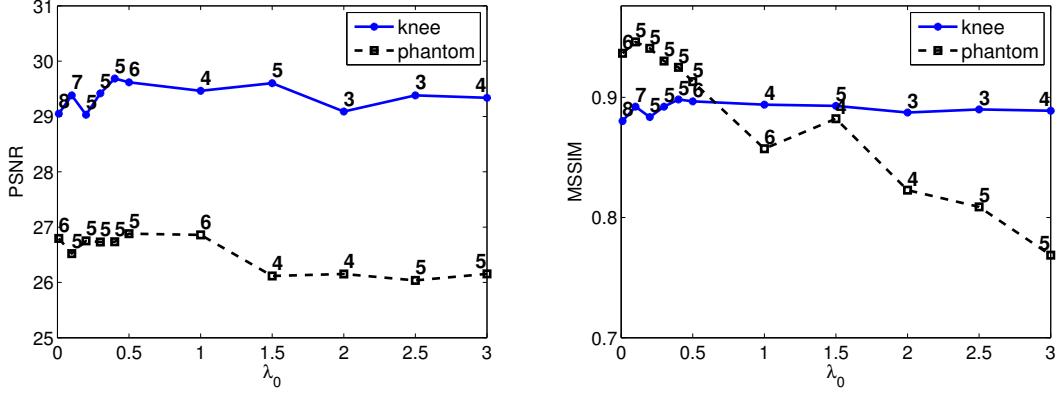
**Figure 8.5:** Final values of  $\lambda$  by our adaptive parameter choice rule.

### 8.3.2. Wavelet Inpainting

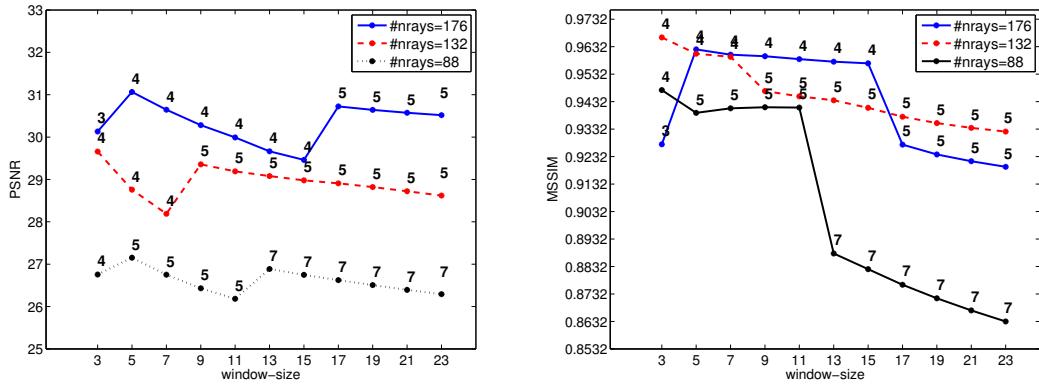
Wavelet inpainting is the problem of filling in missing data in the wavelet domain. This data may be lost by conventional communication techniques, which cannot provide an error free transmission. Hence, firstly a certain amount of wavelet coefficients is missing and secondly there might be some noise added to the data. In our experiments we consider random loss of data, which we specify with the amount of data available by its ratio. For example, if ratio = 0.8, then 80% of the data is available or in other words, 20% of the original data got lost. Additionally, the observed data is contaminated by Gaussian noise with zero mean and standard deviation  $\sigma$ .

As above we study the performance of the sSA-TV method and compare its results with the restorations of the primal-dual Newton method. For the latter method we again perform many experiments with different scalar  $\lambda$ -values and present here the results with the largest

## II. Parameter Selection Methods for Total Variation Models

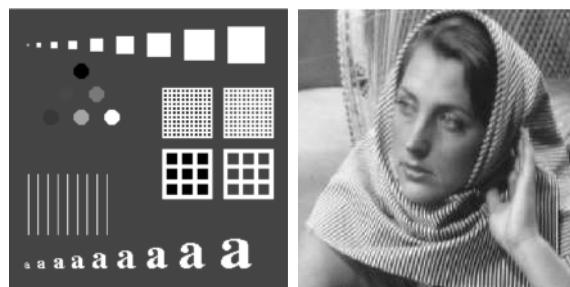


**Figure 8.6:** PSNR and MSSIM for the “phantom”- and “knee”-image restored from partial Fourier data (sampled along 88 radial lines), which is distorted by additive Gaussian noise with zero mean and standard deviation  $\sigma = 0.1$  by our method for different  $\lambda_0$ .



**Figure 8.7:** Restoration of the phantom-image with  $\sigma = 0.1$  via the sSA-TV method for different  $\omega$ .

PSNR- and MSSIM-values.



**Figure 8.8:** Images obtained by full Wavelet-coefficients (a) “phantom” (b) “barbara”.

## 8. Adaptive Regularization for Parseval Frames in Image Processing

$\sigma$	ratio	primal dual Newton		SA-TV method		sSA-TV method	
		PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM
0.3	0.9	19.11 ( $\lambda = 5$ )	0.671 ( $\lambda = 1$ )	19.55	0.774	<b>19.97</b>	<b>0.855</b>
0.3	0.8	17.91 ( $\lambda = 5$ )	0.645 ( $\lambda = 1$ )	18.29	0.750	<b>18.54</b>	<b>0.823</b>
0.3	0.7	17.36 ( $\lambda = 5$ )	0.643 ( $\lambda = 5$ )	17.52	0.672	<b>18.18</b>	<b>0.749</b>
0.3	0.6	16.16 ( $\lambda = 5$ )	0.642 ( $\lambda = 5$ )	16.38	0.649	<b>16.89</b>	<b>0.733</b>
0.3	0.5	14.72 ( $\lambda = 5$ )	0.626 ( $\lambda = 5$ )	15.22	0.613	<b>15.79</b>	<b>0.669</b>
0.1	0.9	26.17 ( $\lambda = 20$ )	0.963 ( $\lambda = 10$ )	25.74	0.900	<b>27.38</b>	<b>0.965</b>
0.1	0.8	24.82 ( $\lambda = 20$ )	0.951 ( $\lambda = 10$ )	23.96	0.895	<b>24.97</b>	<b>0.952</b>
0.1	0.7	<b>23.18</b> ( $\lambda = 20$ )	0.929 ( $\lambda = 10$ )	22.00	0.883	22.87	<b>0.930</b>
0.1	0.6	<b>21.31</b> ( $\lambda = 20$ )	0.873 ( $\lambda = 10$ )	20.22	0.824	20.59	<b>0.894</b>
0.1	0.5	18.54 ( $\lambda = 20$ )	0.787 ( $\lambda = 20$ )	18.28	0.785	<b>18.75</b>	<b>0.843</b>

**Table 8.3:** For the “phantom”-image we test for different standard deviations  $\sigma$  and different ratios of known wavelet coefficients for coiflet-1 wavelet. The solution of the total variation minimization with a constant regularization parameter is compared with the solution of the adaptive total variation approaches (SA-TV and sSA-TV) with  $\lambda_0 = 0.3$  and  $\omega = 11$  in the sense of the quality measures PSNR and MSSIM.

For our investigation we consider the two images shown in Figure 8.8 both of size  $256 \times 256$  pixels. We test our sSA-TV method for different noise-levels, i.e.,  $\sigma \in \{0.3, 0.1\}$ , different ratios, i.e., 0.9, 0.8, 0.7, 0.6, and 0.5, and different wavelets, i.e., coiflet-1 and Daubechies-6. The obtained PSNR- and MSSIM-values of the solution generated by the sSA-TV method with  $\omega = 11$  and  $\lambda_0 = 0.3$  and  $\lambda_0 = 1$ , respectively, are summarized in Table 8.3, Table 8.4, Table 8.5 and Table 8.6.

From these results we observe that the sSA-TV method outperforms the primal-dual Newton method with respect to MSSIM. With respect to PSNR, the sSA-TV method also performs better than the primal-dual Newton method in most of the cases and especially when the ratio is large, i.e., when a lot of data is available. For example, when we consider a random loss of 20% of the wavelet coefficients (i.e., ratio = 0.8) and  $\sigma = 0.05$ , we see in Figure 8.9 that the sSA-TV method preserves image features (see the pattern on the scarf in Figure 8.9), while removing noise in the uniform parts. This is due to the fact, that the adaptively computed function  $\lambda$  is much higher in regions where details (e.g., see the pattern on the scarf) have to be reconstructed, see Figure 8.9 (c). The primal-dual Newton method leaves noise in the image while destroying image features, see Figure 8.9 (d)-(e) and (g)-(h). A similar behaviour is observed for the example in Figure 8.10 where  $\sigma = 0.1$  and ratio = 0.5. Here we clearly see, that the primal-dual Newton method with a scalar regularization parameter is not able to preserve the intensities in the image, e.g., the light gray of the squares.

We also compared the results of the sSA-TV algorithm with the ones of the SA-TV algorithm concerning the image in Figure 8.8(a). From Table 8.3 we observe that the sSA-TV algorithm is superior to the SA-TV algorithm with respect to PSNR and MSSIM. For  $\sigma = 0.1$  the results of the SA-TV method have even smaller PSNR- and MSSIM-values as the ones generated by the primal-dual Newton method with scalar  $\lambda$ , while for  $\sigma = 0.3$  we observe the opposite, i.e., PSNR and MSSIM of the results obtained by the SA-TV algorithm are larger than the ones of the primal-dual Newton method.

As above for the reconstruction of partial Fourier-data the final adaptive parameter  $\lambda$  obtained by our parameter choice rule is indeed chosen according to the features in the image, see

## II. Parameter Selection Methods for Total Variation Models

$\sigma$	ratio	primal dual Newton		sSA-TV method	
		PSNR	MSSIM	PSNR	MSSIM
0.3	0.9	19.72 ( $\lambda = 5$ )	0.464 ( $\lambda = 5$ )	<b>20.34</b>	<b>0.522</b>
0.3	0.8	19.57 ( $\lambda = 5$ )	0.462 ( $\lambda = 5$ )	<b>19.95</b>	<b>0.501</b>
0.3	0.7	19.17 ( $\lambda = 5$ )	0.453 ( $\lambda = 5$ )	<b>19.32</b>	<b>0.477</b>
0.3	0.6	<b>18.80</b> ( $\lambda = 5$ )	0.442 ( $\lambda = 5$ )	<b>18.80</b>	<b>0.453</b>
0.3	0.5	15.75 ( $\lambda = 10$ )	0.381 ( $\lambda = 5$ )	<b>17.52</b>	<b>0.387</b>
0.1	0.9	23.51 ( $\lambda = 20$ )	0.719 ( $\lambda = \frac{100}{7}$ )	<b>24.03</b>	<b>0.756</b>
0.1	0.8	22.63 ( $\lambda = 20$ )	0.686 ( $\lambda = \frac{100}{7}$ )	<b>22.93</b>	<b>0.718</b>
0.1	0.7	21.62 ( $\lambda = 20$ )	0.645 ( $\lambda = 20$ )	<b>21.64</b>	<b>0.667</b>
0.1	0.6	<b>20.78</b> ( $\lambda = 20$ )	0.610 ( $\lambda = 20$ )	20.61	<b>0.614</b>
0.1	0.5	<b>19.86</b> ( $\lambda = \frac{100}{7}$ )	0.559 ( $\lambda = \frac{100}{7}$ )	19.61	<b>0.571</b>

**Table 8.4:** For the “barbara”-image we test for different standard deviations  $\sigma$  and different ratios of known wavelet coefficients for coiflet-1 wavelet. The solution of the total variation minimization with a constant regularization parameter is compared with the solution of the adaptive total variation approach with  $\lambda_0 = 1$  and  $\omega = 11$  in the sense of the quality measures PSNR and MSSIM.

$\sigma$	ratio	primal dual Newton		sSA-TV method	
		PSNR	MSSIM	PSNR	MSSIM
0.3	0.9	18.86 ( $\lambda = 5$ )	0.641 ( $\lambda = 1$ )	<b>19.94</b>	<b>0.857</b>
0.3	0.8	18.30 ( $\lambda = 5$ )	0.636 ( $\lambda = 1$ )	<b>18.77</b>	<b>0.824</b>
0.3	0.7	16.99 ( $\lambda = 5$ )	0.620 ( $\lambda = 5$ )	<b>17.59</b>	<b>0.787</b>
0.3	0.6	16.29 ( $\lambda = 5$ )	0.631 ( $\lambda = 5$ )	<b>16.53</b>	<b>0.749</b>
0.3	0.5	15.42 ( $\lambda = 5$ )	0.626 ( $\lambda = 5$ )	<b>15.85</b>	<b>0.672</b>
0.1	0.9	25.66 ( $\lambda = 20$ )	0.953 ( $\lambda = 10$ )	<b>26.38</b>	<b>0.964</b>
0.1	0.8	24.50 ( $\lambda = 20$ )	0.937 ( $\lambda = 10$ )	<b>25.05</b>	<b>0.951</b>
0.1	0.7	21.95 ( $\lambda = 20$ )	0.909 ( $\lambda = 10$ )	<b>22.01</b>	<b>0.021</b>
0.1	0.6	20.12 ( $\lambda = 20$ )	0.873 ( $\lambda = 10$ )	<b>20.27</b>	<b>0.892</b>
0.1	0.5	18.43 ( $\lambda = 20$ )	0.822 ( $\lambda = 10$ )	<b>18.61</b>	<b>0.849</b>

**Table 8.5:** For the “phantom”-image we test for different standard deviations  $\sigma$  and different ratios of known wavelet coefficients for Daubechies-6 wavelet. The solution of the total variation minimization with a constant regularization parameter is compared with the solution of the adaptive total variation approach with  $\lambda_0 = 0.3$  and  $\omega = 11$  in the sense of the quality measures PSNR and MSSIM.

$\sigma$	ratio	primal dual Newton		sSA-TV method	
		PSNR	MSSIM	PSNR	MSSIM
0.3	0.9	19.63 ( $\lambda = 5$ )	0.454 ( $\lambda = 5$ )	<b>20.28</b>	<b>0.515</b>
	0.8	19.55 ( $\lambda = 5$ )	0.458 ( $\lambda = 5$ )	<b>20.04</b>	<b>0.506</b>
	0.7	19.08 ( $\lambda = 5$ )	0.441 ( $\lambda = 5$ )	<b>19.42</b>	<b>0.477</b>
	0.6	18.71 ( $\lambda = 5$ )	0.435 ( $\lambda = 5$ )	<b>18.89</b>	<b>0.456</b>
	0.5	18.09 ( $\lambda = 5$ )	0.416 ( $\lambda = 5$ )	<b>18.16</b>	<b>0.417</b>
0.1	0.9	23.06 ( $\lambda = 20$ )	0.688 ( $\lambda = 20$ )	<b>23.86</b>	<b>0.747</b>
	0.8	22.30 ( $\lambda = 20$ )	0.665 ( $\lambda = 20$ )	<b>22.95</b>	<b>0.707</b>
	0.7	21.25 ( $\lambda = 20$ )	0.628 ( $\lambda = 20$ )	<b>21.63</b>	<b>0.647</b>
	0.6	20.28 ( $\lambda = 20$ )	0.594 ( $\lambda = 20$ )	<b>20.53</b>	<b>0.595</b>
	0.5	<b>19.24</b> ( $\lambda = 20$ )	<b>0.551</b> ( $\lambda = 20$ )	19.19	0.534

**Table 8.6:** For the ‘‘barbara’’-image we test for different standard deviations  $\sigma$  and different ratios of known wavelet coefficients for Daubechies-6 wavelet. The solution of the total variation minimization with a constant regularization parameter is compared with the solution of the adaptive total variation approach with  $\lambda_0 = 1$  and  $\omega = 11$  in the sense of the quality measures PSNR and MSSIM.

Figure 8.9(c) and Figure 8.10(c). We see that in detail regions  $\lambda$  is large in order to preserve details, while in uniform parts it is small to remove noise considerably.

### 8.3.2.1. Dependence on the Initial Choice $\lambda_0$

Now we investigate our algorithm concerning its stability with respect to  $\lambda_0$  for the application of wavelet inpainting. We test the sSA-TV method for  $\lambda_0 \in \{0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4\}$  and plot the PSNR- and MSSIM-values obtained for the reconstruction in Figure 8.11. In these plots we also specify the number of updates on  $\lambda$  for each  $\lambda_0$ . We observe a similar behaviour as in the case of MRI. Although now the reconstructions seem to be much more stable with respect to  $\lambda_0$ .

### 8.3.2.2. Dependence on the Window Size $\omega$

As for the reconstruction from partial Fourier-data we test our algorithm again for different values of the window size  $\omega$  varying from 3 to 23. In Figure 8.12 and Figure 8.13 we plot the PSNR- and MSSIM-values of the restoration of the image from Figure 8.10 and Figure 8.9 obtained by the sSA-TV method with  $\lambda_0 = 0.3$  and  $\lambda_0 = 1$ , respectively. We observe that the reconstruction is remarkably stable with respect to the window size for all considered random losses of data.

## 8.4. Conclusion

In this work it has been shown that spatially adaptive data fidelity weights help to improve the quality of restored images with respect to PSNR and MSSIM. For undersampled data, this, of course, depends on the sampling rate, i.e. in case the sampling operation induces a loss of details in the data, then these details cannot be recovered properly by adjusting fidelity weights only. On the other hand, as long as the sampling allows to detect details

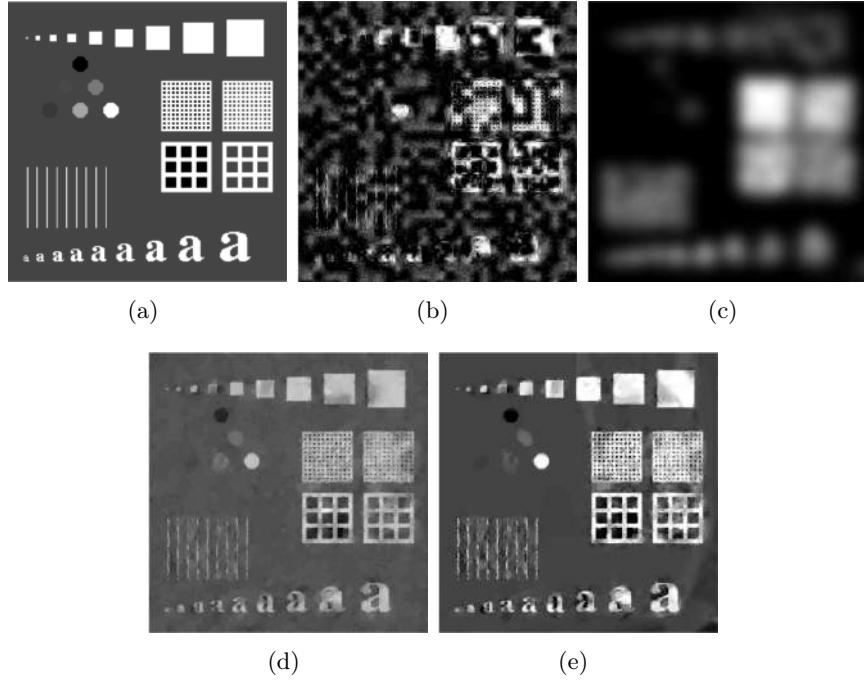
## II. Parameter Selection Methods for Total Variation Models



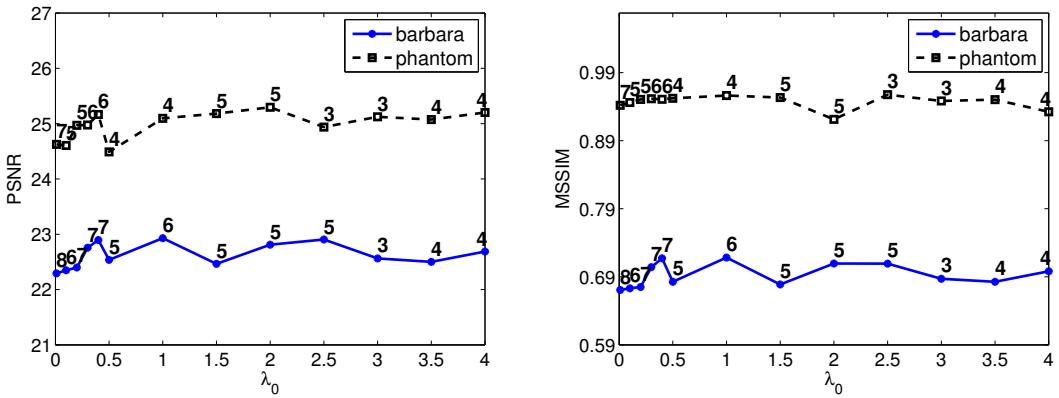
**Figure 8.9:** Random loss of 20% of wavelet coefficients with Gaussian noise  $\sigma = 0.1$  (a) Original image before information loss (b) Received image back-projected (c) Final values of  $\lambda$  by our adaptive parameter choice rule (d) Best PSNR-reconstruction by the primal-dual Newton algorithm ( $\lambda = 20$ , PSNR=22.63, MSSIM=0.676) (e) Best MSSIM-reconstruction by the primal-dual Newton algorithm ( $\lambda = \frac{100}{7}$ , PSNR=22.58, MSSIM=0.686) (f) Reconstruction via sSA-TV method (PSNR=22.93, MSSIM=0.718). In (g) - (i) we zoomed on the scarf.

in images, adaptive fidelity weights outperform best experimental scalar choices with respect to the aforementioned quality measures. In our context, the automated adjustment of the local weights is based on the localized image residuals and confidence regions. It has the advantage of avoiding over/under-fitting due to properly chosen bounds in confidence tests. Algorithmically, the parameter adjustment scheme can be sped up by employing hierarchical decompositions, which aim at decomposing an image into so-called “atoms” at different scales, where a scale is induced by the magnitude of the fidelity weight. The framework of the paper is suitable for data transfer operators  $K$ , which relate to Parseval frames, or when  $K$

## 8. Adaptive Regularization for Parseval Frames in Image Processing



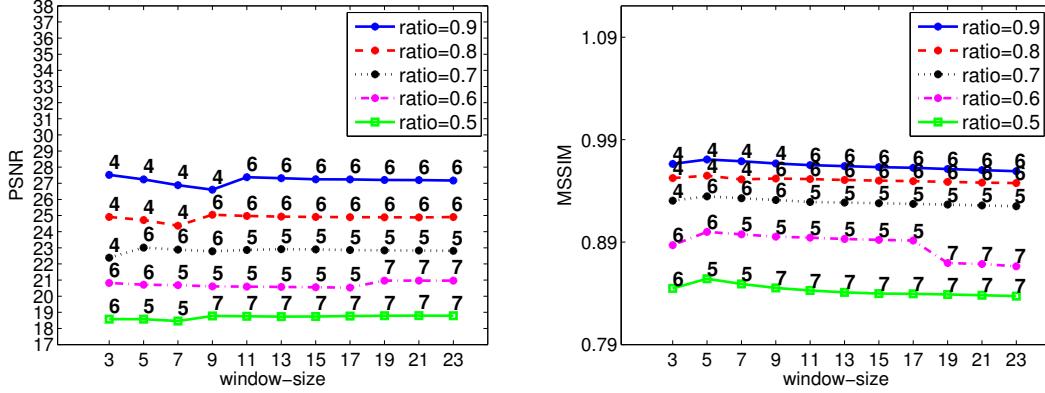
**Figure 8.10:** Random loss of 50% of wavelet coefficients with Gaussian noise  $\sigma = 0.1$  (a) Original image before information loss (b) Received image back-projected (c) Final values of  $\lambda$  by our adaptive parameter choice rule (d) Best reconstruction by the primal-dual Newton algorithm ( $\lambda = 20$ ,  $PSNR=18.54$ ,  $MSSIM=0.787$ ) (e) Reconstruction via sSATV method ( $PSNR=18.75$ ,  $MSSIM=0.843$ ).



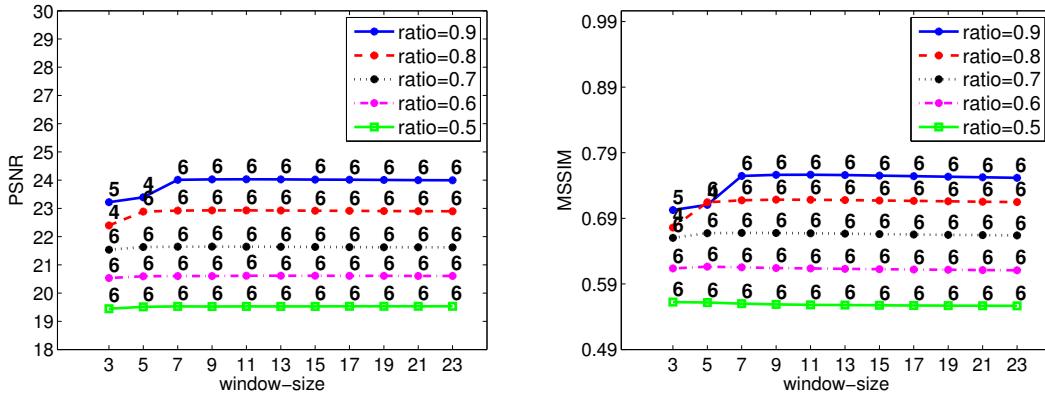
**Figure 8.11:** PSNR and MSSIM for the “phantom”- and “barbara”-image restored from partial wavelet coefficients (ratio = 0.8), which is distorted by additive Gaussian noise with zero mean and standard deviation  $\sigma = 0.1$ , by our method for different  $\lambda_0$ .

encodes a blurring type operation. This is due to the structure of the output space and the properties of the adjoint operator  $K^*$ . In the context of Parseval frames, wavelet inpainting

## II. Parameter Selection Methods for Total Variation Models



**Figure 8.12:** Restoration of the image from Figure 8.10 with  $\sigma = 0.1$  via the sSA-TV method for different  $\omega$ .



**Figure 8.13:** Restoration of the image from Figure 8.9 with  $\sigma = 0.1$  via the sSA-TV method for different  $\omega$ .

or recovery from partial Fourier data represent two specific applications, which can be treated by the framework developed in this paper. Allowing for more general operators  $K$  and, hence, perhaps transferring from spatially adaptive data fidelity weights to distributed regularization weights remains a future challenge.

## 8.5. References

- [1] R. A. Adams and J. J. F. Fournier. *Sobolev Spaces*, volume 140 of *Pure and Applied Mathematics (Amsterdam)*. Elsevier/Academic Press, Amsterdam, second edition, 2003.
- [2] A. Almansa, C. Ballester, V. Caselles, and G. Haro. A TV based restoration model with local constraints. *J. Sci. Comput.*, 34(3):209–236, 2008.
- [3] M. Bertalmío, V. Caselles, B. Rougé, and A. Solé. TV based image restoration with local constraints. *Journal of Scientific Computing*, 19(1-3):95–122, 2003.

## 8. Adaptive Regularization for Parseval Frames in Image Processing

- [4] A. C. Bovik. *Handbook of Image and Video Processing*. Academic press, 2010.
- [5] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.
- [6] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97, 2004.
- [7] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, 1997.
- [8] T. F. Chan, J. Shen, and H.-M. Zhou. Total variation wavelet inpainting. *Journal of Mathematical Imaging and Vision*, 25(1):107–125, 2006.
- [9] Q. Chang and I.-L. Chern. Acceleration methods for total variation-based image denoising. *SIAM Journal on Scientific Computing*, 25(3):982–994, 2003.
- [10] K. Chen, E. L. Piccolomini, and F. Zama. An automatic regularization parameter selection algorithm in the total variation model for image deblurring. *Numerical Algorithms*, 67(1):73–92, 2014.
- [11] I. Daubechies. *Ten lectures on wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. SIAM, Philadelphia, PA, 1992.
- [12] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communication on Pure and Applied Mathematics*, 57(11):1413–1457, 2004.
- [13] I. Daubechies, G. Teschke, and L. Vese. Iteratively solving linear inverse problems under general convex constraints. *Inverse Problems and Imaging*, 1(1):29–46, 2007.
- [14] D. C. Dobson and C. R. Vogel. Convergence of an iterative method for total variation denoising. *SIAM Journal on Numerical Analysis*, 34(5):1779–1791, 1997.
- [15] Y. Dong, M. Hintermüller, and M. M. Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *Journal of Mathematical Imaging and Vision*, 40(1):82–104, 2011.
- [16] K. Frick, P. Marnitz, and A. Munk. Statistical multiresolution dantzig estimation in imaging: fundamental concepts and algorithmic framework. *Electronic Journal of Statistics*, 6:231–268, 2012.
- [17] K. Frick, P. Marnitz, and A. Munk. Statistical multiresolution estimation for variational imaging: With an application in poisson-biophotonics. *Journal of Mathematical Imaging and Vision*, 46(3):370–387, 2013.
- [18] R. Giryes, M. Elad, and Y. C. Eldar. The projected GSURE for automatic parameter tuning in iterative shrinkage methods. *Applied and Computational Harmonic Analysis*, 30(3):407–422, 2011.
- [19] T. Goldstein and S. Osher. The split Bregman method for  $L_1$ -regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.
- [20] C. He, C. Hu, W. Zhang, and B. Shi. A fast adaptive parameter estimation for total variation image restoration. *IEEE Transactions on Image Processing*, 23(12):4954–4967, 2014.
- [21] M. Hintermüller and K. Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM Journal on Applied Mathematics*, 64(4):1311–1333, 2004.
- [22] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.

## II. Parameter Selection Methods for Total Variation Models

- [23] T. Hotz, P. Marnitz, R. Stichtenoth, L. Davies, Z. Kabluchko, and A. Munk. Locally adaptive image denoising by a statistical multiresolution criterion. *Computational Statistics & Data Analysis*, 56(3):543–558, 2012.
- [24] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.
- [25] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.
- [26] D. Strong and T. Chan. Edge-preserving and scale-dependent properties of total variation regularization. *Inverse Problems*, 19(6):165–187, 2003.
- [27] D. M. Strong, J.-F. Aujol, and T. F. Chan. Scale recognition, regularization parameter selection, and Meyer’s  $G$  norm in total variation regularization. *Multiscale Modeling & Simulation*, 5(1):273–303, 2006.
- [28] D. M. Strong and T. F. Chan. Spatially and scale adaptive total variation based regularization and anisotropic diffusion in image processing. In *Division in Image Processing, UCLA Math Department CAM Report*. Citeseer, 1996.
- [29] C. Sutour, C.-A. Deledalle, and J.-F. Aujol. Adaptive regularization of the NL-means: Application to image and video denoising. *IEEE Transactions on Image Processing*, 23(8):3506–3521, 2014.
- [30] E. Tadmor, S. Nezzar, and L. Vese. A multiscale image representation using hierarchical  $(BV, L^2)$  decompositions. *Multiscale Modeling & Simulation*, 2(4):554–579, 2004.
- [31] E. Tadmor, S. Nezzar, and L. Vese. Multiscale hierarchical decomposition of images with applications to deblurring, denoising and segmentation. *Commun. Math. Sci.*, 6(2):281–307, 2008.
- [32] C. R. Vogel. *Computational Methods for Inverse Problems*, volume 23. SIAM, 2002.
- [33] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
- [34] Y.-W. Wen and R. H. Chan. Parameter selection for total-variation-based image restoration using discrepancy principle. *IEEE Transactions on Image Processing*, 21(4):1770–1781, 2012.
- [35] X. Zhang and T. F. Chan. Wavelet inpainting by nonlocal total variation. *Inverse problems and Imaging*, 4(1):191–210, 2010.

## 9. Influence of the Regularization Parameter on Box-Constrained Total Variation

**Bibliographic Note:** The content of this chapter is published in [AL8]: A. Langer. Investigating the influence of box-constraints on the solution of a total variation model via an efficient primal-dual method. *Journal of Imaging*, 4(1):12, 2018.

**Summary:** In this paper, we investigate the usefulness of adding a box-constraint to the minimization of functionals consisting of a data-fidelity term and a total variation regularization term. In particular, we show that in certain applications an additional box-constraint does not effect the solution at all, i.e., the solution is the same whether a box-constraint is used or not. On the contrary, i.e., for applications where a box-constraint may have influence on the solution, we investigate how much it effects the quality of the restoration, especially when the regularization parameter, which weights the importance of the data term and the regularizer, is chosen suitable. In particular, for such applications, we consider the case of a squared  $L^2$  data-fidelity term. For computing a minimizer of the respective box-constrained optimization problems a primal-dual semismooth Newton method is presented, which guarantees superlinear convergence.

### 9.1. Introduction

An observed image  $g$ , which contains additive Gaussian noise with zero mean and standard deviation  $\sigma$ , may be modeled as

$$g = K\hat{u} + n$$

where  $\hat{u}$  is the original image,  $K$  is a linear bounded operator and  $n$  represents the noise. With the aim of preserving edges in images in [61] total variation regularization in image restoration was proposed. Based on this approach and assuming that  $g \in L^2(\Omega)$  and  $K \in \mathcal{L}(L^2(\Omega))$ , a good approximation of  $\hat{u}$  is usually obtained by solving

$$\min_{u \in BV(\Omega)} \int_{\Omega} |Du| \quad \text{subject to (s.t.)} \quad \|Ku - g\|_{L^2(\Omega)}^2 \leq \sigma^2 |\Omega| \quad (9.1)$$

where  $\Omega \subset \mathbb{R}^2$  is a simply connected domain with Lipschitz boundary and  $|\Omega|$  its volume. Here  $\int_{\Omega} |Du|$  denotes the total variation of  $u$  in  $\Omega$  and  $BV(\Omega)$  is the space of functions with bounded variation, i.e.,  $u \in BV(\Omega)$  if and only if  $u \in L^1(\Omega)$  and  $\int_{\Omega} |Du| < \infty$ ; see [3, 38] for more details. We recall, that  $BV(\Omega) \subset L^2(\Omega)$ , if  $\Omega \subset \mathbb{R}^2$ .

Instead of considering (9.1), we may solve the penalized minimization problem

$$\min_{u \in BV(\Omega)} \|Ku - g\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} |Du| \quad (9.2)$$

## II. Parameter Selection Methods for Total Variation Models

for a given constant  $\alpha > 0$ , which we refer to the  $L^2$ -TV model. In particular, there exists a constant  $\alpha \geq 0$  such that the constrained problem (9.1) is equivalent to the penalized problem (9.2), if  $g \in K(BV(\Omega))$  and  $K$  does not annihilate constant functions [16]. Moreover, under the latter condition also the existence of a minimizer of (9.1) and (9.2) is guaranteed [16]. There exist many algorithms that solve (9.1) and (9.2), see for example [7, 9, 10, 13, 15, 17, 20, 21, 22, 23, 24, 25, 26, 30, 40, 41, 45, 56, 60] and references therein.

If in (9.2) instead of the quadratic  $L^2$ -norm an  $L^1$ -norm is used, we refer to it as the  $L^1$ -TV model. The quadratic  $L^2$ -norm is usually used when Gaussian noise is contained in the image, while the  $L^1$ -norm is more suitable for impulse noise [2, 58, 59].

If we additionally know (or assume) that the original image lies in the dynamic range  $[c_{\min}, c_{\max}]$ , i.e.,  $c_{\min} \leq u(x) \leq c_{\max}$  for almost every (a.e.)  $x \in \Omega$ , we incorporate this information into our model (9.1) and (9.2) leading to

$$\min_{u \in BV(\Omega)} \int_{\Omega} |Du| \quad \text{s.t.} \quad \|Ku - g\|_{L^2(\Omega)}^2 \leq \sigma^2 |\Omega| \quad \text{and} \quad u \in C \quad (9.3)$$

and

$$\min_{u \in BV(\Omega) \cap C} \|Ku - g\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} |Du|, \quad (9.4)$$

respectively, where  $C := \{u \in L^2(\Omega) : c_{\min} \leq u(x) \leq c_{\max} \text{ for a.e. } x \in \Omega\}$ . In order to guarantee the existence of a minimizer of (9.3) and (9.4) we assume in the sequel that  $K$  does not annihilate constant functions<sup>1</sup>. If additionally  $g \in K(BV(\Omega) \cap C)$ , then by [16, Prop. 2.1] it follows that there exists a constant  $\alpha \geq 0$  such that (9.3) is equivalent to (9.4).

For image restoration box-constraints have been considered for example in [7, 19, 53, 55]. In [55] a functional consisting of an  $L^2$ -data term and a Tikhonov-like regularization term (i.e.,  $L^2$ -norm of some derivative of  $u$ ) in connection with box-constrained is presented together with a Newton-like numerical scheme. For box-constrained total variation minimization in [7] a fast gradient-based algorithm, called monotone fast iterative shrinkage/thresholding algorithm (MFISTA), is proposed and a rate of convergence is proven. Based on the alternating direction method of multipliers (ADMM) [36] in [19] a solver for the box-constrained  $L^2$ -TV and  $L^1$ -TV model is derived and shown to be faster than MFISTA. In [53] a primal-dual algorithm for the box-constrained  $L^1$ -TV model and for box-constrained nonlocal total variation is presented. In order to achieve a constrained solution, which is positive and bounded from above by some intensity value, in [66] an exponential type transform is applied to the  $L^2$ -TV model. Recently, in [52] a box-constraint is also incorporated in a total variation model with a combined  $L^1$ - $L^2$  data fidelity, proposed in [42], for removing simultaneously Gaussian and impulse noise in images.

Setting the upper bound in the set  $C$  to infinity and the lower bound to 0, i.e.,  $c_{\min} = 0$  and  $c_{\max} = +\infty$ , leads to a nonnegativity constraint. Total variation minimization with a nonnegativity constraint is a well-known technique to improve the quality of reconstructions in image processing; see for example [5, 63] and references therein.

In this paper we are concerned with (9.3) and (9.4) when the lower bound  $c_{\min}$  and the upper bound  $c_{\max}$  in  $C$  are finite. However, the analysis and the presented algorithms are easily adjustable to the situation when one of the bounds is set to  $-\infty$  or  $+\infty$  respectively. Note, that a solution of (9.1) and (9.2) is in general not an element of the set  $C$ . However,

---

<sup>1</sup>By noting that the characteristic function  $\chi_C$  is lower semicontinuous this follows by the same arguments as in [16].

since  $g$  is an observation containing Gaussian noise with zero mean, a minimizer of (9.2) lies indeed in  $C$ , if  $\alpha$  in (9.2) is sufficiently large and the original image  $\hat{u} \in C$ . This observation however rises the question whether an optimal parameter  $\alpha$  would lead to a minimizer that lies in  $C$ . If this would be the case then incorporating the box-constraint into the minimization problem does not gain any improvement of the solution. In particular, there are situations in which a box-constraint is not effecting the solution at all (see Section 9.3 below). Additionally we expect that the box-constrained problems are more difficult to handle and numerically more costly to solve than (9.1) and (9.2).

In order to answer the above raised question, we numerically compute optimal values of  $\alpha$  for the box-constrained total variation and the non-box-constrained total variation and compare the resulting reconstructions with respect to quality measures. By optimal values we mean here parameters  $\alpha$  such that the solutions of (9.1) and (9.2) or (9.3) and (9.4) coincide. Note, that there exists several methods for computing the regularization parameter; see for example [48] for an overview of parameter selection algorithms for image restoration. Here we use the pAPS-algorithm proposed in [48] to compute reasonable  $\alpha$  in (9.2) and (9.4). For minimizing (9.4) we derive a semismooth Newton method, which should serve us as a good method for quickly computing rather exact solutions. Second order methods have been already proposed and used in image reconstruction; see [31, 32, 44, 45, 48]. However, to the best of our knowledge till now semismooth Newton methods have not been presented for box-constrained total variation minimization. In this setting, differently to the before mentioned approaches, the box-constraint adds some additional difficulties in deriving the dual problems, which have to be calculated to obtain the desired method; see Section 9.4 for more details. The superlinear convergence of our method is guaranteed by the theory of semismooth Newton methods; see for example [45]. Note, that our approach differs significantly from the Newton-like scheme presented in [55], where a smooth objective functional with a box-constraint is considered. This allows in [55] to derive a Newton method without dualization. Here, our Newton method is based on dualization and may be viewed as a primal-dual (Newton) approach.

We remark, that a scalar regularization parameter might not be the best choice for every image restoration problem, since images usually consist of large uniform areas and parts with fine details, see for example [32, 48]. It has been demonstrated, for example in [14, 32, 48, 50] and references therein, that with the help of spatially varying regularization parameters one might be able to restore images visually better than with scalar parameters. In this vein we also consider

$$\min_{u \in BV(\Omega)} \|Ku - g\|_{L^2(\Omega)}^2 + \int_{\Omega} \alpha |Du| \quad (9.5)$$

and

$$\min_{u \in BV(\Omega) \cap C} \|Ku - g\|_{L^2(\Omega)}^2 + \int_{\Omega} \alpha |Du|, \quad (9.6)$$

where  $\alpha : \Omega \rightarrow \mathbb{R}^+$  is a bounded continuous function [4]. We adapt our semismooth Newton method to approximately solve these two optimization problems and utilize the pLATV-algorithm of [48] to compute a locally varying  $\alpha$ .

Our numerical results show, see Section 9.6, that in a lot of applications the quality of the restoration is more a question of how to choose the regularization parameter then including a box-constraint. However, the solutions obtained by solving the box-constrained versions (9.3), (9.4) and (9.6) are improving the restorations slightly, but not drastically. Nevertheless, we also report on a medical applications where a nonnegativity constraint significantly improves the restoration.

## II. Parameter Selection Methods for Total Variation Models

We realize, that if the noise-level of the corrupted image is unknown, then we may use the information of the image intensity range (if known) to calculate a suitable parameter for problem (9.2). Note, that in this situation the formulations (9.1) and (9.3) cannot be considered since  $\sigma$  is not at hand. We present a method which automatically computes the regularization parameter  $\alpha$  in (9.2) provided the information that the original image  $\hat{u} \in [c_{\min}, c_{\max}]$ .

Hence the contribution of the paper is three-sided: (i) We present a semismooth Newton method for the box-constrained total variation minimization problems (9.3) and (9.6). (ii) We investigate the influence of the box-constraint on the solution of the total variation minimization models with respect to the regularization parameter. (iii) In case the noise-level is not at hand, we propose a new automatic regularization parameter selection algorithm based on the box-constraint information.

The outline of the rest of the paper is organized as follows: In Section 9.2 we recall useful definitions and the Fenchel-duality theorem which will be used later. Section 9.3 is devoted to the analysis of the box-constrained total variation minimization. In particular, we state that in certain cases adding a box-constraint to the considered problem does not change the solution at all. The semismooth Newton method for the box-constrained  $L^2$ -TV model (9.4) and its multiscale version (9.6) is derived in Section 9.4 and its numerical implementation is presented in Section 9.5. Numerical experiments investigating the usefulness of a box-constraint are shown in Section 9.6. In Section 9.7 we propose an automatic parameter selection algorithm by using the box-constraint. Finally, in Section 9.8 conclusions are drawn.

### 9.2. Basic Terminology

Let  $X$  be a Banach space. Its topological dual is denoted by  $X^*$  and  $\langle \cdot, \cdot \rangle$  describes the bilinear canonical pairing over  $X \times X^*$ . A convex functional  $J : X \rightarrow \bar{\mathbb{R}}$  is called *proper*, if  $\{v \in X : J(v) \neq +\infty\} \neq \emptyset$  and  $J(v) > -\infty$  for all  $v \in X$ . A functional  $J : X \rightarrow \bar{\mathbb{R}}$  is called *lower semicontinuous*, if for every weakly convergent sequence  $v^{(n)} \rightharpoonup \hat{v}$  we have

$$\liminf_{v^{(n)} \rightharpoonup \hat{v}} J(v^{(n)}) \geq J(\hat{v}).$$

For a convex functional  $J : X \rightarrow \bar{\mathbb{R}}$  we define the *subdifferential* of  $J$  at  $v \in X$  as the set valued function

$$\partial J(v) := \begin{cases} \emptyset & \text{if } J(v) = \infty, \\ \{v^* \in X^* : \langle v^*, u - v \rangle + J(v) \leq J(u) \quad \forall u \in X\} & \text{otherwise.} \end{cases}$$

It is clear from this definition, that  $0 \in \partial J(v)$  if and only if  $v$  is a minimizer of  $J$ .

The *conjugate function (or Legendre transform)* of a convex function  $J : X \rightarrow \bar{\mathbb{R}}$  is defined as  $J^* : X^* \rightarrow \bar{\mathbb{R}}$  with

$$J^*(v^*) = \sup_{v \in X} \{\langle v, v^* \rangle - J(v)\}.$$

From this definition we see that  $J^*$  is the pointwise supremum of continuous affine functions and thus, according to [34, Proposition 3.1, p 14], convex, lower semicontinuous, and proper.

For an arbitrary set  $S$  we denote by  $\chi_S$  its characteristic function defined by

$$\chi_S(u) = \begin{cases} 0 & \text{if } u \in S, \\ \infty & \text{otherwise.} \end{cases}$$

We recall the Fenchel duality theorem; see, e.g., [34] for details.

**Theorem 9.1** (Fenchel duality theorem). *Let  $X$  and  $Y$  be two Banach spaces with topological duals  $X^*$  and  $Y^*$ , respectively, and  $\Lambda : X \rightarrow Y$  a bounded linear operator with adjoint  $\Lambda^* \in \mathcal{L}(Y^*, X^*)$ . Further let  $\mathcal{F} : X \rightarrow \mathbb{R} \cup \{\infty\}$ ,  $\mathcal{G} : Y \rightarrow \mathbb{R} \cup \{\infty\}$  be convex, lower semicontinuous, and proper functionals. Assume there exists  $u_0 \in X$  such that  $\mathcal{F}(u_0) < \infty$ ,  $\mathcal{G}(\Lambda u_0) < \infty$  and  $\mathcal{G}$  is continuous at  $\Lambda u_0$ . Then we have*

$$\inf_{u \in X} \mathcal{F}(u) + \mathcal{G}(\Lambda u) = \sup_{p \in Y^*} -\mathcal{F}^*(\Lambda^* p) - \mathcal{G}^*(-p) \quad (9.7)$$

and the problem on the right hand side of (9.7) admits a solution  $\bar{p}$ . Moreover,  $\bar{u}$  and  $\bar{p}$  are solutions of the two optimization problems in (9.7), respectively, if and only if

$$\begin{aligned} \Lambda^* \bar{p} &\in \partial \mathcal{F}(\bar{u}), \\ -\bar{p} &\in \partial \mathcal{G}(\Lambda \bar{u}). \end{aligned}$$

### 9.3. Limitation of Box-Constrained Total Variation Minimization

In this section we investigate the difference between the box-constrained problem (9.4) and the non-box-constrained problem (9.1). For the case when the operator  $K$  is the identity  $I$ , which is the relevant case in image denoising, we have the following obvious result:

**Proposition 9.2.** *Let  $K = I$  and  $g \in C$ , then the minimizer  $u^* \in BV(\Omega)$  of (9.2) lies also in the dynamic range  $[c_{\min}, c_{\max}]$ , i.e.,  $u^* \in BV(\Omega) \cap C$ .*

*Proof of Proposition 9.2.* Assume  $u^* \in BV(\Omega) \setminus C$  is a minimizer of (9.2). Define a function  $\tilde{u}$  such that

$$\tilde{u}(x) := \begin{cases} u^*(x) & \text{if } c_{\min} \leq u^*(x) \leq c_{\max}, \\ c_{\max} & \text{if } u^*(x) > c_{\max}, \\ c_{\min} & \text{if } u^*(x) < c_{\min}, \end{cases}$$

for a.e.  $x \in \Omega$ . Then we have that

$$\|u^* - g\|_{L^2(\Omega)} > \|\tilde{u} - g\|_{L^2(\Omega)}, \text{ and } \int_{\Omega} |Du^*| > \int_{\Omega} |\tilde{D}\tilde{u}|. \quad (9.8)$$

This implies that  $u^*$  is not a minimizer of (9.2), which is a contradiction and hence  $u^* \in BV(\Omega) \cap C$ .  $\square$

This result is easily extendable to optimization problems of the type

$$\min_{u \in BV(\Omega)} \alpha_1 \|u - g\|_{L^1(\Omega)} + \alpha_2 \|u - g\|_{L^2(\Omega)}^2 + \int_{\Omega} |Du|, \quad (9.9)$$

with  $\alpha_1, \alpha_2 \geq 0$  and  $\alpha_1 + \alpha_2 > 0$ , since for  $\tilde{u}$ , defined as in the above proof, and a minimizer  $u^* \in BV(\Omega) \cap C$  of (9.9) the inequalities in (9.8) hold as well as  $\|u^* - g\|_{L^1(\Omega)} > \|\tilde{u} - g\|_{L^1(\Omega)}$ . Problem (9.9) has been already considered in [1, 42, 49] and can be viewed as a generalization of the  $L^2$ -TV model, since  $\alpha_1 = 0$  in (9.9) yields the  $L^2$ -TV model and  $\alpha_2 = 0$  in (9.9) yields the  $L^1$ -TV model.

## II. Parameter Selection Methods for Total Variation Models

Note, that if an image is only corrupted by impulse noise, then the observed image  $g$  is in the dynamic range of the original image. For example, salt-and-pepper noise contained images may be written as

$$g(x) = \begin{cases} c_{\min} & \text{with probability } s_1 \in [0, 1), \\ c_{\max} & \text{with probability } s_2 \in [0, 1), \\ \hat{u}(x) & \text{with probability } 1 - s_1 - s_2, \end{cases}$$

with  $1 - s_1 - s_2 > 0$  [18] and for *random-valued impulse noise*  $g$  is described as

$$g(x) = \begin{cases} d & \text{with probability } s \in [0, 1), \\ \hat{u}(x) & \text{with probability } 1 - s, \end{cases}$$

with  $d$  being a uniformly distributed random variable in the image intensity range  $[c_{\min}, c_{\max}]$ . Hence, following Proposition 9.2, in such cases considering constrained total variation minimization would not change the minimizer and no improvement in the restoration quality can be expected.

This is the reason why we restrict ourselves in the rest of the paper to Gaussian white noise contaminated images and consider solely the  $L^2$ -TV model.

It is clear that if a solution of the non box-constrained optimization problem already fulfills the box-constraint, then it is of course equivalent to a minimizer of the box-constraint problem. However, note that the minimizer is not unique in general.

In the following we compare the solution of the box-constrained optimization problem (9.4) with the solution of the unconstrained minimization problem (9.2).

**Proposition 9.3.** *Let  $u \in C \cap BV(\Omega)$  be a minimizer of*

$$J^C(u) := \frac{1}{2} \|u - g\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} |Du| + \chi_C(u)$$

and  $w \in BV(\Omega)$  be a minimizer of

$$J(w) := \frac{1}{2} \|w - g\|_{L^2(\Omega)}^2 + \alpha |Dw|(\Omega).$$

Then we have that

1.  $J^C(w) \geq J^C(u) = J(u) \geq J(w)$ .
2.  $\frac{1}{2} \|u - w\|_{L^2(\Omega)}^2 \leq J(u) - J(w) \leq J^C(w) - J^C(u)$ .
3.  $\|u - w\|_{L^2(\Omega)}^2 \leq 4\|\xi - g\|_{L^2(\Omega)}^2 + 8\alpha|D\xi|(\Omega)$  for any  $\xi \in C \cap BV(\Omega)$ .

*Proof of Proposition 9.3.*

1. Follows directly from the optimality of  $u$  and  $w$ .
2. From [6, Lemma 10.2] it follows that  $\frac{1}{2} \|u - w\|_{L^2(\Omega)}^2 \leq J(u) - J(w)$ . For the second inequality we make the observation that

$$J^C(w) - J^C(u) = \begin{cases} \infty & \text{if } w \notin C \\ 0 & \text{if } w \in C, \end{cases}$$

where we used the fact that  $w = u$  if  $w \in C$ . This implies, that  $J(u) - J(w) \leq J^C(w) - J^C(u)$ .

3. For all  $v \in C \cap BV(\Omega)$  we have that

$$\begin{aligned}\|u - w\|_{L^2(\Omega)}^2 &\leq 2 \left( \|u - v\|_{L^2(\Omega)}^2 + \|v - w\|_{L^2(\Omega)}^2 \right) \\ &\leq 4(J(v) - J(u) + J(v) - J(w)) = 8J(v) - 4J(u) - 4J(w),\end{aligned}$$

where we used 2 and that  $(a+b)^2 \leq 2(a^2 + b^2)$ . For any arbitrary  $\xi \in C \cap BV(\Omega)$ , let  $v = \xi$  and since  $J(\xi) = \frac{1}{2}\|\xi - g\|_{L^2(\Omega)}^2 + \alpha|D\xi|(\Omega)$  we get  $\|u - w\|_{L^2(\Omega)}^2 \leq 4\|\xi - g\|_{L^2(\Omega)}^2 + 8\alpha|D\xi|(\Omega)$ .

□

**Remark 9.4.** If in Proposition 9.3  $\xi \in C \cap BV(\Omega)$  is constant, then  $|D\xi|(\Omega) = 0$  which implies that  $\|u - w\|_{L^2(\Omega)}^2 \leq 4\|\xi - g\|_{L^2(\Omega)}^2$ .

## 9.4. A Semismooth Newton Method

### 9.4.1. The Model Problem

In general  $K^*K$  is not invertible, which causes difficulties in deriving the dual problem of (9.4). In order to overcome this difficulties we penalize the  $L^2$ -TV model by considering the following neighboring problem

$$\min_{u \in C \cap H_0^1(\Omega)} \frac{1}{2}\|Ku - g\|_{L^2(\Omega)}^2 + \frac{\mu}{2} \int_{\Omega} |\nabla u|_{\ell^2}^2 dx + \alpha \int_{\Omega} |\nabla u|_{\ell^2} dx, \quad (9.10)$$

where  $\mu > 0$  is a very small constant such that (9.10) is a close approximation of the total variation regularized problem (9.4). Note, that for  $u \in H_0^1(\Omega)$  the total variation of  $u$  in  $\Omega$  is equivalent to  $\int_{\Omega} |\nabla u|_{\ell^2} dx$  [38]. A typical example for which  $K^*K$  is indeed invertible is  $K = I$ , which is used for image denoising. In this case we may even set  $\mu = 0$ , see Section 9.6. The functional in (9.10) has been already considered for example in [46, 45] for image restoration. In particular in [45], a primal-dual semismooth Newton algorithm is introduced. Here, we actually adopt this approach to our box-constrained problem (9.10).

In the sequel we assume for simplicity that  $-c_{\min} = c_{\max} =: c > 0$ , which changes the set  $C$  to  $C := \{u \in L^2(\Omega) : |u| \leq c\}$ . Note, that any bounded image  $\hat{u}$ , i.e., which lies in the dynamic range  $[a, b]$ , can be easily transformed to an image  $\tilde{u} \in [-c, c]$ . Since this transform and  $K$  are linear, the observation  $g$  is also easily transformed to  $\tilde{g} = K\tilde{u} + n$ .

**Example 9.5.** Let  $\hat{u}$  such that  $a \leq \hat{u}(x) \leq b$  for all  $x \in \Omega$ . Then  $|\hat{u}(x) - \frac{b+a}{2}| \leq \frac{b-a}{2} =: c$  for all  $x \in \Omega$  and we set  $\tilde{u} = \hat{u} - \frac{b+a}{2}$ . Hence,  $\tilde{g} = K\hat{u} - K\frac{b+a}{2} + n = g - K\frac{b+a}{2}$ .

Problem (9.10) can be equivalently written as

$$\min_{u \in H_0^1(\Omega)} \frac{1}{2}\|Ku - g\|_{L^2(\Omega)}^2 + \frac{\mu}{2} \int_{\Omega} |\nabla u|_{\ell^2}^2 dx + \alpha \int_{\Omega} |\nabla u|_{\ell^2} dx + \chi_C(u). \quad (9.11)$$

If  $u^* \in H_0^1(\Omega)$  is a solution of (9.10) (and equivalently (9.11)), then there exists  $\lambda^* \in H_0^1(\Omega)^*$  and  $\sigma^* \in \partial R(u)$ , where  $R(u) := \int_{\Omega} |\nabla u|_{\ell^2} dx$ , such that

$$\begin{aligned}K^*Ku^* - K^*g - \mu\Delta u^* + \alpha\sigma^* + \lambda^* &= 0 \\ \langle \lambda^*, u - u^* \rangle &\leq 0\end{aligned}$$

## II. Parameter Selection Methods for Total Variation Models

for all  $u \in C \cap H_0^1(\Omega)$ .

For implementation reasons (actually for obtaining a fast, second-order algorithm) we approximate the nonsmooth characteristic function  $\chi_C$  by a smooth function in the following way

$$\begin{aligned}\chi_C(u) &\approx \frac{\eta}{2} \left( \|\max\{u - c_{\max}, 0\}\|_{L^2(\Omega)}^2 + \|\max\{c_{\min} - u, 0\}\|_{L^2(\Omega)}^2 \right) \\ &= \frac{\eta}{2} \left( \|\max\{|u| - c, 0\}\|_{L^2(\Omega)}^2 \right),\end{aligned}$$

where  $\eta > 0$  is large. This leads to the following optimization problem

$$\min_{u \in H_0^1(\Omega)} \frac{1}{2} \|Ku - g\|_{L^2(\Omega)}^2 + \frac{\mu}{2} \int_{\Omega} |\nabla u|_{\ell^2}^2 dx + \alpha \int_{\Omega} |\nabla u|_{\ell^2} dx + \frac{\eta}{2} \left( \|\max\{|u| - c, 0\}\|_{L^2(\Omega)}^2 \right). \quad (9.12)$$

**Remark 9.6.** By the assumption  $-c_{\min} = c_{\max}$  we actually excluded the cases (i)  $c_{\min} = 0$ ,  $c_{\max} = +\infty$  and (ii)  $c_{\min} = -\infty$ ,  $c_{\max} = 0$ . In these situations we just need to approximate  $\chi_C$  by (i)  $\frac{\eta}{2} \left( \|\max\{-u, 0\}\|_{L^2(\Omega)}^2 \right)$  and (ii)  $\frac{\eta}{2} \left( \|\max\{u, 0\}\|_{L^2(\Omega)}^2 \right)$ . By noting this, in a similar fashion as done below for the problem in (9.12), a primal-dual semismooth Newton method can be derived for these two cases.

### 9.4.2. Dualization

By a standard calculation one obtains that the dual of (9.12) is given by

$$\sup_{\mathbf{p}=(\mathbf{p}_1,p_2) \in \mathbb{L}^2(\Omega) \times L^2(\Omega)} -\frac{1}{2} \|\Lambda^* \mathbf{p} + K^* g\|_B^2 + \frac{1}{2} \|g\|_{L^2(\Omega)}^2 - \chi_A(-\mathbf{p}_1) - \frac{1}{2\eta} \|-p_2\|_{L^2(\Omega)}^2 - \|-cp_2\|_{L^1(\Omega)} \quad (9.13)$$

with  $\Lambda^* \mathbf{p} = -\operatorname{div} \mathbf{p}_1 + p_2$  and  $A := \{\mathbf{v} \in \mathbb{L}^2(\Omega) : |\mathbf{v}|_{\ell^2} \leq \alpha\}$ . As the divergence operator does not have a trivial kernel, the solution of the optimization problem (9.13) is not unique. In order to render the problem (9.13) strictly concave we add an additional term yielding the following problem

$$\begin{aligned}\min_{\mathbf{p} \in \mathbb{L}^2(\Omega) \times L^2(\Omega)} &\frac{1}{2} \|\Lambda^* \mathbf{p} + K^* g\|_B^2 - \frac{1}{2} \|g\|_{L^2(\Omega)}^2 + \chi_A(-\mathbf{p}_1) + \frac{1}{2\eta} \|-p_2\|_{L^2(\Omega)}^2 \\ &+ \|-cp_2\|_{L^1(\Omega)} + \frac{\gamma}{2\alpha} \|p_1\|_{L^2(\Omega)}^2,\end{aligned} \quad (9.14)$$

where  $\gamma > 0$  is a fixed parameter.

**Proposition 9.7.** The dual problem of (9.14) is given by

$$\min_{u \in H_0^1(\Omega)} \frac{1}{2} \|Ku - g\|_{L^2(\Omega)}^2 + \frac{\mu}{2} \|\nabla u\|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} \phi_{\gamma}(\nabla u)(x) dx + \frac{\eta}{2} \|\max\{|u| - c, 0\}\|_{L^2(\Omega)}^2 \quad (9.15)$$

with

$$\phi_{\gamma}(\mathbf{q})(x) = \begin{cases} \frac{1}{2\gamma} |\mathbf{q}(x)|_{\ell^2}^2 & \text{if } |\mathbf{q}(x)|_{\ell^2} < \gamma \\ |\mathbf{q}(x)|_{\ell^2} - \frac{\gamma}{2} & \text{if } |\mathbf{q}(x)|_{\ell^2} \geq \gamma. \end{cases}$$

## 9. Box-Constraint Total Variation

*Proof.* In order to compute the Fenchel dual of (9.14) we set  $q = -p$ ,

$$\begin{aligned}\mathcal{F}(\mathbf{q}) &:= \chi_A(\mathbf{q}_1) + \frac{\gamma}{2\alpha} \|\mathbf{q}_1\|_{L^2(\Omega)}^2 + \frac{1}{2\eta} \|q_2\|_{L^2(\Omega)}^2 + \|cq_2\|_{L^1(\Omega)} \\ \mathcal{G}(\Lambda\mathbf{q}) &:= \frac{1}{2} \||K^*g - \Lambda\mathbf{q}||_B^2 - \frac{1}{2} \|g\|_{L^2(\Omega)}^2, \quad \Lambda\mathbf{q} := q_2 - \operatorname{div} \mathbf{q}_1,\end{aligned}$$

with  $X = \mathbb{L}^2(\Omega) \times L^2(\Omega)$  and  $Y = H_0^1(\Omega)^* = H^{-1}(\Omega)$ .

By the definition of conjugate we have

$$\mathcal{G}^*(u^*) = \sup_{u \in Y} \{ \langle u, u^* \rangle - \frac{1}{2} \langle B(K^*g - u), K^*g - u \rangle + \frac{1}{2} \|g\|_{L^2(\Omega)}^2 \}.$$

Then  $u$  is a supremum if

$$\partial_u \{ \langle u, u^* \rangle - \mathcal{G}(u) \} = u^* + B(K^*g - u) = 0,$$

which implies  $u = B^{-1}u^* + 2\alpha_2 T_2^* g_2$ . Hence

$$\begin{aligned}\mathcal{G}^*(u^*) &= \langle B^{-1}u^* + K^*g, u^* \rangle - \frac{1}{2} \langle BB^{-1}u^*, B^{-1}u^* \rangle + \frac{1}{2} \|g\|_{L^2(\Omega)}^2 \\ &= \langle u^*, B^{-1}u^* \rangle + \langle u^*, K^*g \rangle - \frac{1}{2} \langle u^*, B^{-1}u^* \rangle + \frac{1}{2} \|g\|_{L^2(\Omega)}^2 \\ &= \frac{1}{2} \langle u^*, (K^*K + \mu \nabla^* \nabla)u^* \rangle + \langle u^*, K^*g \rangle + \frac{1}{2} \|g\|_{L^2(\Omega)}^2 \\ &= \frac{1}{2} \langle Ku^*, Ku^* \rangle + \frac{\mu}{2} \langle \nabla u^*, \nabla u^* \rangle + \langle Ku^*, g \rangle + \frac{1}{2} \|g\|_{L^2(\Omega)}^2 \\ &= \frac{1}{2} \|Ku^* + g\|_{L^2(\Omega)}^2 + \frac{\mu}{2} \|\nabla u^*\|_{L^2(\Omega)}^2.\end{aligned}$$

In order to compute the conjugate  $\mathcal{F}^*$  we split  $\mathcal{F}$  into two functionals  $\mathcal{F}_1$  and  $\mathcal{F}_2$  defined as

$$\mathcal{F}_1(\mathbf{q}_1) := \chi_A(\mathbf{q}_1) + \frac{\gamma}{2\alpha} \|\mathbf{q}_1\|_{L^2(\Omega)}^2, \quad \mathcal{F}_2(q_2) := \frac{1}{2\eta} \|q_2\|_{L^2(\Omega)}^2 + \|cq_2\|_{L^1(\Omega)},$$

whereas  $\mathcal{F}^*(\mathbf{q}^*) = \mathcal{F}_1^*(\mathbf{q}_1^*) + \mathcal{F}_2^*(q_2^*)$ . We have, that

$$\mathcal{F}_1(\mathbf{q}_1^*) = \sup_{\mathbf{q}_1 \in \mathbb{L}^2(\Omega)} \{ \langle \mathbf{q}_1, \mathbf{q}_1^* \rangle - \chi_A(\mathbf{q}_1) - \frac{\gamma}{2\alpha} \|\mathbf{q}_1\|_{L^2(\Omega)}^2 \}.$$

A function  $\mathbf{q}_1$  is a supremum of this set if

$$\mathbf{q}_1^* - \frac{\gamma}{\alpha} \mathbf{q}_1 = 0$$

with  $|\mathbf{q}_1|_{\ell^2} \leq \alpha$ . The equality implies  $\mathbf{q}_1 = \frac{\alpha}{\gamma} \mathbf{q}_1^*$  from which we deduce

$$\mathcal{F}_1^*(\mathbf{q}_1^*)(x) = \begin{cases} \frac{\alpha}{2\gamma} |\mathbf{q}_1^*(x)|_{\ell^2}^2 & \text{if } |\mathbf{q}_1^*(x)|_{\ell^2} < \gamma, \\ \alpha |\mathbf{q}_1^*(x)|_{\ell^2} - \frac{\alpha\gamma}{2} & \text{if } |\mathbf{q}_1^*(x)|_{\ell^2} \geq \gamma. \end{cases}$$

For the conjugate  $\mathcal{F}_2^*$  of  $\mathcal{F}_2$  we get

$$\mathcal{F}_2^*(q_2^*) = \sup_{q_2 \in L^2(\Omega)} \{ \langle q_2, q_2^* \rangle - \frac{1}{2\eta} \|q_2\|_{L^2(\Omega)}^2 - \|cq_2\|_{L^1(\Omega)} \}.$$

## II. Parameter Selection Methods for Total Variation Models

Hence  $q_2$  is a supremum if

$$q_2^* - \frac{1}{\eta} q_2 - c\sigma_{\|\cdot\|_1} = 0 \quad \text{with } \sigma_{\|\cdot\|_1} \in \partial \|cq_2\|_{L^1(\Omega)}. \quad (9.16)$$

Thus

$$\begin{aligned} \mathcal{F}_2^*(\mathbf{q}_2^*) &= \langle \eta q_2^* - \eta c\sigma_{\|\cdot\|_1}, q_2^* \rangle - \frac{1}{2\eta} \|\eta q_2^* - \eta c\sigma_{\|\cdot\|_1}\|_{L^2(\Omega)}^2 - \|c\eta q_2^* - c^2\eta\sigma_{\|\cdot\|_1}\|_{L^1(\Omega)} \\ &= \eta \langle q_2^* - \eta c\sigma_{\|\cdot\|_1}, q_2^* - \eta c\sigma_{\|\cdot\|_1} \rangle + \eta \langle q_2^* - \eta c\sigma_{\|\cdot\|_1}, \eta c\sigma_{\|\cdot\|_1} \rangle - \frac{\eta}{2} \|\eta q_2^* - \eta c\sigma_{\|\cdot\|_1}\|_{L^2(\Omega)}^2 \\ &\quad - \|c\eta q_2^* - c^2\eta\sigma_{\|\cdot\|_1}\|_{L^1(\Omega)} \\ &= \frac{\eta}{2} \|q_2^* - c\sigma_{\|\cdot\|_1}\|_{L^2(\Omega)}^2 + \eta \int_{\{q_2 \geq 0\}} (q_2^* - c)c - |cq_2^* - c^2| dx \\ &\quad + \eta \int_{\{q_2 < 0\}} (q_2^* + c)(-c) - |cq_2^* + c^2| dx. \end{aligned}$$

From (9.16) we obtain that

$$\begin{aligned} &\text{if } q_2 = 0 \text{ then } q_2^* = c\sigma_{\|\cdot\|_1}, \\ &\text{if } q_2 > 0 \text{ then } q_2^* > c, \\ &\text{if } q_2 < 0 \text{ then } q_2^* < c. \end{aligned}$$

Using this observation yields

$$\begin{aligned} \mathcal{F}_2^*(q_2^*) &= \frac{\eta}{2} \|q_2^* - c\sigma_{\|\cdot\|_1}\|_{L^2(\Omega)}^2 = \frac{\eta}{2} \int_{\{q_2 \geq 0\}} |q_2^* - c|^2 dx + \int_{\{q_2 < 0\}} |q_2^* + c|^2 dx \\ &= \frac{\eta}{2} \|\max\{|q_2^*| - c, 0\}\|_{L^2(\Omega)}^2. \end{aligned}$$

By the Fenchel duality theorem the assertion follows.  $\square$

Similar as in [45] one can show that the solution of (9.15) converges to the minimizer of (9.12) as  $\gamma \rightarrow 0$ .

From the Fenchel duality theorem we obtain the following characterization of solutions  $u$  and  $\mathbf{p}$  of (9.15) and (9.14)

$$\operatorname{div} \mathbf{p}_1 - p_2 = K^*Ku - K^*g - \mu\Delta u \quad \text{in } H^{-1}(\Omega) \quad (9.17)$$

$$\mathbf{p}_1 = \frac{\alpha}{\gamma} \nabla u \quad \text{if } |\mathbf{p}_1|_{\ell^2} < \alpha \quad \text{in } \mathbb{L}^2(\Omega) \quad (9.18)$$

$$\mathbf{p}_1 = \alpha \frac{\nabla u}{|\nabla u|_{\ell^2}} \quad \text{if } |\mathbf{p}_1|_{\ell^2} = \alpha_1 \quad \text{in } \mathbb{L}^2(\Omega) \quad (9.19)$$

$$p_2 = \eta \max\{|u| - c, 0\} \operatorname{sign}(u) \quad \text{in } L^2(\Omega). \quad (9.20)$$

This system can be solved efficiently by a semismooth Newton algorithm. Moreover, equation (9.18) and (9.19) can be condensed into  $\mathbf{p}_1 = \frac{\alpha \nabla u}{\max\{\gamma, |\nabla u|_{\ell^2}\}}$ .

### 9.4.3. Adaptation to Nonscalar $\alpha$

For locally adaptive  $\alpha$ , i.e.,  $\alpha : \Omega \rightarrow \mathbb{R}^+$  is a function, the minimization problem (9.12) changes to

$$\min_{u \in H_0^1(\Omega)} \frac{1}{2} \|Ku - g\|_{L^2(\Omega)}^2 + \frac{\mu}{2} \int_{\Omega} |\nabla u|_{\ell^2}^2 dx + \int_{\Omega} \alpha(x) |\nabla u|_{\ell^2} dx + \frac{\eta}{2} \left( \|\max\{|u| - c, 0\}\|_{L^2(\Omega)}^2 \right). \quad (9.21)$$

Its dual problem is given by

$$\min_{\mathbf{p} \in \mathbb{L}^2(\Omega) \times L^2(\Omega)} \frac{1}{2} \|\Lambda^* \mathbf{p} + K^* g\|_B^2 - \frac{1}{2} \|g\|_{L^2(\Omega)}^2 + \chi_{\tilde{A}}(-\mathbf{p}_1) + \frac{1}{2\eta} \|-p_2\|_{L^2(\Omega)}^2 + \|-cp_2\|_{L^1(\Omega)},$$

where  $\tilde{A} := \{\mathbf{v} \in \mathbb{L}^2(\Omega) : |\mathbf{v}(x)|_{\ell^2} \leq \alpha(x) \text{ for a.e. } x \in \Omega\}$ . Similarly but slightly different as above, cf. (9.14), we penalize by

$$\begin{aligned} \min_{\mathbf{p} \in \mathbb{L}^2(\Omega) \times L^2(\Omega)} & \frac{1}{2} \|\Lambda^* \mathbf{p} + K^* g\|_B^2 - \frac{1}{2} \|g\|_{L^2(\Omega)}^2 + \chi_A(-\mathbf{p}_1) + \frac{1}{2\eta} \|-p_2\|_{L^2(\Omega)}^2 \\ & + \|-cp_2\|_{L^1(\Omega)} + \frac{\gamma}{2} \|p_1\|_{L^2(\Omega)}^2. \end{aligned}$$

Then the dual of this problem turns out to be

$$\min_{u \in H_0^1(\Omega)} \frac{1}{2} \|Ku - g\|_{L^2(\Omega)}^2 + \frac{\mu}{2} \|\nabla u\|_{L^2(\Omega)}^2 + \int_{\Omega} \phi_{\gamma, \alpha}(\nabla u)(x) dx + \frac{\eta}{2} \|\max\{|u| - c, 0\}\|_{L^2(\Omega)}^2 \quad (9.22)$$

with

$$\phi_{\gamma, \alpha}(\mathbf{q})(x) = \begin{cases} \frac{1}{2\gamma} |\mathbf{q}(x)|_{\ell^2}^2 & \text{if } |\mathbf{q}(x)|_{\ell^2} < \gamma\alpha(x) \\ \alpha(x) |\mathbf{q}(x)|_{\ell^2} - \frac{\gamma}{2} |\alpha(x)|^2 & \text{if } |\mathbf{q}(x)|_{\ell^2} \geq \gamma\alpha(x). \end{cases}$$

Denoting by  $u$  a solution of (9.22) and  $\mathbf{p}$  a solution of the associated pre-dual problem, the optimality conditions due to the Fenchel theorem [34] are given by

$$\begin{aligned} \operatorname{div} \mathbf{p}_1 - p_2 &= K^* Ku - K^* g - \mu \Delta u \\ \mathbf{p}_1 &= \frac{\alpha \nabla u}{\max\{\gamma\alpha, |\nabla u|_{\ell^2}\}} \\ p_2 &= \eta \max\{|u| - c, 0\} \operatorname{sign}(u). \end{aligned}$$

## 9.5. Numerical Implementation

Similar as in the works [31, 32, 44, 45, 48], where semismooth Newton methods for nonsmooth systems emerging from image restoration models have been derived, we can solve the discrete version of the system (9.17)-(9.20), using finite differences, efficiently by a primal-dual algorithm. Therefore let  $u^h \in \mathbb{R}^N$ ,  $p_1^h \in \mathbb{R}^{2N}$ ,  $p_2^h \in \mathbb{R}^N$ ,  $g^h \in \mathbb{R}^N$ , denote the discrete image intensity, the dual variables, and the observed data vector, respectively, where  $N \in \mathbb{N}$  is the number of elements (pixels) in the discrete image  $\Omega^h$ . Moreover, we denote by  $\alpha^h > 0$  the regularization parameter. Correspondingly we define  $\nabla^h \in \mathbb{R}^{2N \times N}$  as the discrete gradient operator,  $\Delta^h \in \mathbb{R}^{N \times N}$  as the discrete Laplace operator,  $K^h \in \mathbb{R}^{N \times N}$  as a discrete operator, and  $(K^h)^t$  its transpose. Moreover  $\operatorname{div}^h = -(\nabla^h)^t$ . Here  $|\cdot|$ ,  $\max\{\cdot, \cdot\}$ , and  $\operatorname{sign}(\cdot)$  are understood for vectors in a component-wise sense. Moreover, we use the function  $[\cdot] : \mathbb{R}^{2N} \rightarrow \mathbb{R}^{2N}$  with  $[\cdot]_i = [\cdot]_{i+N} = \sqrt{(v_i^h)^2 + (v_{i+N}^h)^2}$  for  $1 \leq i \leq N$ .

## II. Parameter Selection Methods for Total Variation Models

### 9.5.1. Scalar $\alpha$

The discrete version of (9.17)-(9.20) reads as

$$\begin{aligned} 0 &= -\operatorname{div}^h p_1^h + \eta D(m_0) + (K^h)^t K^h u^h - (K^h)^t g^h - \mu \Delta^h u^h \\ 0 &= D^h(m_\gamma) p_1^h - \alpha^h \nabla^h u^h \end{aligned} \quad (9.23)$$

where  $D^h(v)$  is a diagonal matrix with vector  $v$  in its diagonal,  $m_0 := \operatorname{sign}(u) \max\{|u| - c, 0\}$ , and  $m_\gamma := \max\{\gamma, [|\nabla^h u^h|]\}$ . We define

$$\begin{aligned} \chi_{\mathcal{A}_\gamma} &= D^h(t_\gamma) \quad \text{with } (t_\gamma)_i = \begin{cases} 0 & \text{if } (m_\gamma)_i = \gamma, \\ 1 & \text{else;} \end{cases} \\ \chi_{\mathcal{A}_c^{\max}} &= D^h(t_c^{\max}) \quad \text{with } (t_c^{\max})_i = \begin{cases} 0 & \text{if } (m_c^{\max})_i = 0, \\ 1 & \text{else;} \end{cases} \\ \chi_{\mathcal{A}_c^{\min}} &= D^h(t_c^{\min}) \quad \text{with } (t_c^{\min})_i = \begin{cases} 0 & \text{if } (m_c^{\min})_i = 0, \\ 1 & \text{else,} \end{cases} \end{aligned}$$

where  $m_c^{\max} := \max\{u - c, 0\}$  and  $m_c^{\min} := \max\{u + c, 0\}$ . Further, we set

$$M^h(v) = \begin{pmatrix} D^h(v_x) & D^h(v_y) \\ D^h(v_x) & D^h(v_y) \end{pmatrix} \quad \text{with } v = (v_x, v_y)^t \in \mathbb{R}^{2N}.$$

Applying a generalized Newton step to solve (9.23) at  $(u_k^h, p_{1,k}^h)$  yields

$$\begin{pmatrix} \eta(\chi_{\mathcal{A}_c^{\max}} + \chi_{\mathcal{A}_c^{\min}}) + (K^h)^t K^h - \mu \Delta^h & -\operatorname{div}^h \\ C_k^h \nabla & D^h(m_\gamma) \end{pmatrix} \begin{pmatrix} \delta_u \\ \delta_{p_1} \end{pmatrix} = \begin{pmatrix} -\mathfrak{F}_1^k \\ -\mathfrak{F}_2^k \end{pmatrix} \quad (9.24)$$

where

$$\begin{aligned} C_k^h &= D^h(p_{1,k}^h) \chi_{\mathcal{A}_\gamma} D^h(m_\gamma)^{-1} M^h(\nabla u_k^h) - \alpha^h D^h(e_{2N}) \\ \mathfrak{F}_1^k &= -\operatorname{div}^h p_{1,k}^h + \eta D^h(m_0) + (K^h)^t K^h u_k^h - (K^h)^t g - \mu \Delta^h u_k^h \\ \mathfrak{F}_2^k &= D^h(m_\gamma) p_{1,k}^h - \alpha^h \nabla^h u_k^h \end{aligned}$$

and  $e_N \in \mathbb{R}^N$  is the identity vector. The diagonal matrix  $D(m_\gamma)$  is invertible, i.e.,

$$\delta_{p_1} = D^h(m_\gamma)^{-1} (-\mathfrak{F}_2^k - C_k^h \nabla \delta_u)$$

and hence we can eliminate  $\delta_{p_1}$  from the Newton system resulting in

$$H_k \delta_u = f_k \quad (9.25)$$

where

$$\begin{aligned} H_k &:= \eta(\chi_{\mathcal{A}_c^{\max}} + \chi_{\mathcal{A}_c^{\min}}) + (K^h)^t K^h - \mu \Delta^h + \operatorname{div}^h D^h(m_\gamma)^{-1} C_k^h \nabla, \\ f_k &:= -\mathfrak{F}_1^k - \operatorname{div}^h D^h(m_\gamma)^{-1} \mathfrak{F}_2^k. \end{aligned}$$

If  $H_k$  is positive definite, then the solution  $\delta_u$  of (9.25) exists and is a descent direction of (9.15). However, in general we cannot expect the positive definiteness of  $H_k$ . In order to ensure

that  $H_k$  is positive definite, we project  $p_{1,k}^h$  onto its feasible set by setting  $((p_{1,k}^h)_i, (p_{1,k}^h)_{i+N})$  to  $\alpha^h \max\{\alpha^h, [|p_{1,k}^h|]_i\}^{-1}((p_{1,k}^h)_i, (p_{1,k}^h)_{i+N})$  for  $i = 1, \dots, N$  which guarantees

$$[|p_{1,k}^h|]_i \leq \alpha^h \quad (9.26)$$

for  $i = 1, \dots, 2N$ . The modified system matrix, denoted by  $H_k^+$ , is then positive definite. Then our semismooth Newton solver may be written as:

**Primal-dual Newton method (pdN):** Initialize  $(u_0^h, p_{1,0}^h) \in \mathbb{R}^N \times \mathbb{R}^{2N}$  and set  $k := 0$ .

1. Determine the active sets  $\chi_{\mathcal{A}_c^{\max}} \in \mathbb{R}^{N \times N}, \chi_{\mathcal{A}_c^{\min}} \in \mathbb{R}^{N \times N}, \chi_{\mathcal{A}_\gamma} \in \mathbb{R}^{N \times N}$ ,
2. If (9.26) is not satisfied, then compute  $H_k^+$ ; otherwise set  $H_k^+ := H_k$ .
3. Solve  $H_k^+ \delta_u = f_k$  for  $\delta_u$ .
4. Compute  $\delta_{p_1}$  by using  $\delta_u$ .
5. Update  $u_{k+1}^h := u_k^h + \delta_u$  and  $p_{1,k+1}^h := p_{1,k}^h + \delta_{p_1}$ .
6. Stop or set  $k := k + 1$  and continue with step 1).

This algorithm converges at a superlinear rate, which follows from standard theory; see [41, 45]. The Newton method is terminated as soon as the initial residual is reduced by a factor of  $10^{-4}$ .

Note, that, since  $\eta = 0$  implies  $p_2 = 0$ , in this case the proposed primal-dual Newton method becomes the method in [45].

### 9.5.2. Nonscalar $\alpha$

A similar semismooth Newton method might be derived for the locally adaptive case by noting that then  $\alpha^h \in \mathbb{R}^N$ , and hence the second equation in (9.23) changes to

$$0 = D^h(m_\gamma)p_1^h - D^h((\alpha^h, \alpha^h)^t)\nabla^h u^h,$$

where  $m_\gamma := \max\{\gamma\alpha^h, |\nabla^h u^h|\}$  leading to (9.24) with

$$C_k^h = D^h(p_{1,k}^h)\chi_{\mathcal{A}_\gamma}D^h(m_\gamma)^{-1}M^h(\nabla u_k^h) - D^h((\alpha^h, \alpha^h)^t)$$

and

$$\mathfrak{F}_2^k = D^h(m_\gamma)p_{1,k}^h - D^h((\alpha^h, \alpha^h)^t)\nabla^h u_k^h.$$

The positive definite modified matrix  $H_k^+$  is then obtained by setting  $((p_{1,k}^h)_i, (p_{1,k}^h)_{i+N})$  to  $\alpha_i^h \max\{\alpha_i^h, [|p_{1,k}^h|]_i\}^{-1}((p_{1,k}^h)_i, (p_{1,k}^h)_{i+N})$  for  $i = 1, \dots, N$ .

## 9.6. Numerical Experiments

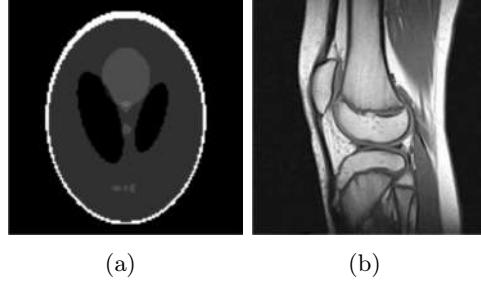
For our numerical studies we consider the images shown in Figure 9.1 of size  $256 \times 256$  pixels, in Figure 9.2, and in Figure 9.15(a). The image intensity range of all original images considered in this paper is  $[0, 1]$ , i.e.,  $c_{\min} = 0$  and  $c_{\max} = 1$ . Our proposed algorithms automatically transform this images into the dynamic range  $[-c, c]$ , here with  $c = 1/2$ . That

## II. Parameter Selection Methods for Total Variation Models

is, let  $\hat{u} \in [0, 1]$  be the original image before any corruption, then  $\hat{u}(x) - \frac{1}{2} \in [-\frac{1}{2}, \frac{1}{2}]$ . Moreover, the solution generated by the semismooth Newton method is afterwards back-transformed, i.e., the generated solution  $\tilde{u}$  is transformed to  $\tilde{u} + \frac{1}{2}$ . Note that  $\max_x \tilde{u}(x) + \frac{1}{2}$  is not necessarily in  $[0, 1]$ , except  $\tilde{u} \in [-\frac{1}{2}, \frac{1}{2}]$ .



**Figure 9.1:** Original images of size  $256 \times 256$



**Figure 9.2:** Original images (a) Shepp-Logan phantom of size  $128 \times 128$  pixels (b) knee of size  $200 \times 200$  pixels

As a comparison for the different restoration qualities of the restored image we use the PSNR [11] (peak signal-to-noise ratio) given by

$$\text{PSNR} = 20 \log \frac{1}{\|\hat{u} - u^*\|},$$

where  $\hat{u}$  denotes the original image before any corruption and  $u^*$  the restored image, which is widely used as an image quality assessment measure, and the MSSIM [64] (mean structural similarity), which usually relates to perceived visual quality better than PSNR. In general,

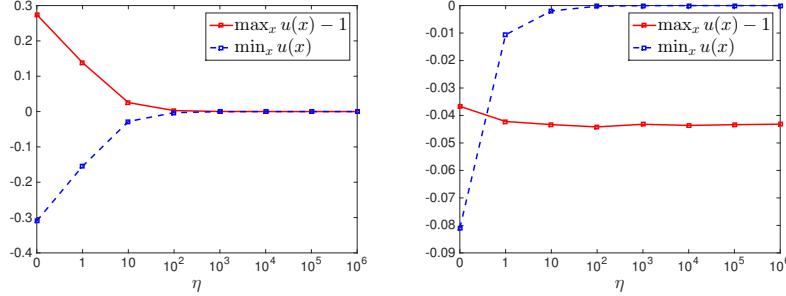
when comparing PSNR and MSSIM, large values indicate better reconstruction than small values.

In our experiments we also report on the computational time (in seconds) and the number of iterations (it) needed until the considered algorithms are terminated.

In all the following experiments the parameter  $\mu$  is chosen to be 0 for image denoising (i.e.,  $K = I$ ), since then no additional smoothing is needed, and  $\mu = 10^{-6}$  if  $K \neq I$  (i.e. for image deblurring, image inpainting, for reconstructing from partial Fourier-data, and for reconstructing from sampled Radon-transform).

### 9.6.1. Dependency on the Parameter $\eta$

We start by investigating the influence of the parameter  $\eta$  on the behavior of the semismooth Newton algorithm and its generated solution. Let us recall, that  $\eta$  is responsible how strictly the box-constraint is adhered. In order to visualize how good the box-constraint is fulfilled for a chosen  $\eta$  in Figure 9.3 we depict  $\max_x u(x) - c_{\max}$  and  $\min_x u(x) - c_{\min}$  with  $c_{\max} = 1$ ,  $c_{\min} = 0$ , and  $u$  being the back-transformed solution, i.e.,  $u = \tilde{u} + \frac{1}{2}$ , where  $\tilde{u}$  is obtained via the semismooth Newton method. As long as  $\max_x u(x) - c_{\max}$  and  $\min_x u(x) - c_{\min}$  are positive and negative, respectively, the box-constraint is not perfectly adhered. From our experiments for image denoising and image deblurring, see Figure 9.3, we clearly see that the larger  $\eta$  the more strictly the box-constraint is adhered. In the rest of our experiments we choose  $\eta = 10^6$ , which seems sufficiently large to us and the box-constraint seems to hold accurately enough.



**Figure 9.3:** Reconstruction of the cameraman image corrupted by Gaussian white noise with  $\sigma = 0.1$  (left), corrupted by blurring and Gaussian white noise with  $\sigma = 0.1$  (right) via the semismooth Newton method with  $\alpha = 0.01$  for different values  $\eta$ .

### 9.6.2. Box-Constrained versus Non-Box-Constrained

In the rest of this section we are going to investigate how much the solution (and its restoration quality) depends on the box-constraint and if this is a matter on how the regularization parameter is chosen. We start by comparing for different values of  $\alpha$  the solutions obtained by the semismooth Newton method without a box-constraint (i.e.  $\eta = 0$ ) with the ones generated by the same algorithm with  $\eta = 10^6$  (i.e. a box-constraint is incorporated). Our obtained results are shown in Table 9.1 for image denoising and in Table 9.2 for image deblurring. We obtain, that for small  $\alpha$  we gain “much” better results with respect to PSNR and MSSIM with a box-constraint than without. The reason for this is that if no box-constraint is used

## II. Parameter Selection Methods for Total Variation Models

and  $\alpha$  is small then nearly no regularization is performed and hence noise, which is violating the box-constraint, is still present. Therefore incorporating a box-constraint is reasonable for these choices of parameters. However, if  $\alpha$  is sufficiently large then we numerically observe that the solution of the box-constrained and non-box-constrained problem are the same. This is not surprising, because there exists  $\bar{\alpha} > 0$  such that for all  $\alpha > \bar{\alpha}$  the solution of (9.2) is  $\frac{1}{|\Omega|} \int_{\Omega} g$ , see [16, Lemma 2.3]. That is, for such  $\alpha$  the minimizer of (9.2) is the average of the observation which lies in the image intensity range of the original image, as long as the mean of Gaussian noise is 0 (or sufficiently small). This implies that in such a case the minimizer of (9.2) and (9.4) are equivalent. Actually this equivalency already holds if  $\alpha$  is sufficiently large such that the respective solution of (9.2) lies in the dynamic range of the original image, which is the case in our experiments for  $\alpha = 0.4$ . Hence, whether it makes sense or not to incorporate a box-constraint into the considered model depends on the choice of parameters. The third and fourth value of  $\alpha$  in Table 9.1 and Table 9.2 refer to the ones which equalize (9.2) and (9.1), and respectively (9.4) and (9.3). For both type of applications, i.e., image denoising and image deblurring, these “optimal”  $\alpha$ -values<sup>2</sup> are nearly the same and hence also the PSNR and MSSIM of the respective result are nearly the same. Nevertheless, we mention that for image deblurring the largest PSNR and MSSIM in these experiments is obtained for  $\alpha = 0.01$  with a box-constraint.

In Table 9.3 and Table 9.4 we also report on an additional strategy. In this approach we threshold (or project) the observation  $g$  such that the box-constraint holds in any pixel and use then the proposed Newton method with  $\eta = 0$ . For large  $\alpha$  this is an inferior approach, but for small  $\alpha$  this seems to work similar to incorporating a box-constraint, at least for image denoising. However, it is outperformed by the other approaches.

$\alpha$	PSNR	pdN with $\eta = 0$			box-constrained pdN		
		MSSIM	time	it	PSNR	MSSIM	time
0.001	20.165	0.27649	183.29	365	20.635	0.28384	178.27
0.01	21.464	0.29712	55.598	117	21.905	0.30472	55.245
0.096029	27.134	0.35214	14.615	33	27.135	0.35221	14.465
0.096108	27.132	0.35201	14.91	33	27.133	0.35207	14.317
0.4	22.079	0.16816	14.779	34	22.079	0.16816	14.982
$\emptyset$	23.5947	0.28919	56.6388	116.4	23.7773	0.2922	55.4557
							116.4

**Table 9.1:** Reconstruction of the cameraman-image corrupted by Gaussian white noise with  $\sigma = 0.1$  for different regularization parameters  $\alpha$ .

### 9.6.3. Comparison with Optimal Regularization Parameters

In order to determine the optimal parameters  $\alpha$  for a range of different examples we assume that the noise-level  $\sigma$  is at hand and utilize the pAPS-algorithm presented in [48]. Alternatively, instead of computing a suitable  $\alpha$ , we may solve the constrained optimization problems in (9.1) and (9.3) directly by using the alternating direction methods of multipliers (ADMM). An implementation of the ADMM for solving (9.1) is presented in [57], which we refer to as the ADMM in the sequel. For solving (9.3) a possible implementation is suggested in [19].

<sup>2</sup>In the sequel we call such parameters *optimal*, since a solution of the penalized problem also solves the related constrained problem. However, we note that these  $\alpha$ -values are in general not giving the best results with respect to PSNR and MSSIM, but they are usually close to the results with the largest PSNR and MSSIM.

$\alpha$	PSNR	pdN with $\eta = 0$			box-constrained pdN		
		MSSIM	time	it	PSNR	MSSIM	time
0.001	5.9375	0.039866	1197	143	11.568	0.083341	1946
0.01	21.908	0.18938	863	67	21.964	0.1946	2011
0.051871	21.815	0.18115	759	37	21.814	0.18113	1299
0.051868	21.814	0.18114	752	36	21.814	0.18114	1255
0.4	19.823	0.090709	1431	61	19.823	0.090709	1454
$\emptyset$	18.2593	0.13645	1000	68.8	19.3966	0.14618	1593
							69.4

**Table 9.2:** Reconstruction of the cameraman-image corrupted by Gaussian blur and Gaussian white noise with  $\sigma = 0.1$  for different regularization parameters  $\alpha$ .

$\alpha$	PSNR	MSSIM	time	it
0.001	20.64	0.28388	107.05	349
0.01	21.949	0.30504	32.336	113
0.096029	26.528	0.33078	8.9479	33
0.096108	26.526	0.33066	8.8966	33
0.4	21.666	0.15794	9.4218	35
$\emptyset$	23.4617	0.28166	33.3304	112.6

**Table 9.3:** Reconstruction of the cameraman-image corrupted by Gaussian white noise with  $\sigma = 0.1$  for different regularization parameters  $\alpha$  using the pdN with thresholded  $g$ .

However, for comparison purposes we use a slightly different version, which uses the same succession of updates as the ADMM in [57], see Section 9.6.3.1 for a description of this version. In the sequel we refer to this algorithm as the box-constrained ADMM. We do not expect the same results for the pAPS-algorithm and the (box-constrained) ADMM, since in the pAPS-algorithm we use the semismooth Newton method which generates an approximate solution of (9.15), that is not equivalent to (9.1) and (9.3). In all the experiments in the pAPS-algorithm we set the initial regularization parameter to be  $10^{-3}$ .

#### 9.6.3.1. Box-Constrained ADMM

In [57] an ADMM for solving the constrained problem (9.1) in a finite dimensional setting is presented. In a similar way we may solve the discrete version of (9.3), i.e.,

$$\min_{u^h \in \mathbb{R}^N} \|\nabla u^h\|_1 \quad \text{s.t.} \quad u^h \in C^h, \quad \frac{1}{N} \|S^h H^h u^h - g^h\|_2^2 \leq \sigma^2, \quad (9.27)$$

$\alpha$	PSNR	MSSIM	time	it
0.001	6.6758	0.046454	1091	140
0.01	21.929	0.19231	743	65
0.051871	21.709	0.1722	659	35
0.051868	21.709	0.1722	651	35
0.4	19.683	0.087447	1438	61
$\emptyset$	18.3413	0.13412	916	67.2

**Table 9.4:** Reconstruction of the cameraman-image corrupted by Gaussian blur and Gaussian white noise with  $\sigma = 0.1$  for different regularization parameters  $\alpha$  using the pdN with thresholded  $g$ .

## II. Parameter Selection Methods for Total Variation Models

where we use the notation of Section 9.5 and  $K^h = S^h H^h$  with  $H^h \in \mathbb{R}^{N \times N}$  being a circular matrix and  $S^h \in \mathbb{R}^{N \times N}$  as in [57]. Moreover,  $C^h := \{u^h \in \mathbb{R}^N : c_{\min} \leq u_i^h \leq c_{\max} \text{ for all } i \in \{1, \dots, N\}\}$ ,  $\|\cdot\|_i$  refers to the standard definition of the  $\ell^i$ -norm, i.e.,  $\|u\|_i := \left(\sum_{j=1}^N |u_j|^i\right)^{\frac{1}{i}}$ , and  $\langle \cdot, \cdot \rangle$  denotes the  $\ell^2$  inner product.

In order to apply the ADMM to problem (9.27) we rewrite it as follows:

$$\begin{aligned} \min_{w^h \in \mathbb{R}^N \times \mathbb{R}^N} \|w^h\|_1 \quad & \text{s.t.} \quad w^h = \nabla^h u, y^h = H^h u, \frac{1}{N} \|S^h y^h - g^h\|_2^2 \leq \nu, \quad z^h = u^h, \\ & z^h \in C^h \end{aligned}$$

which is equivalent to

$$\begin{aligned} \min_{w^h \in \mathbb{R}^N \times \mathbb{R}^N, y^h, z^h \in \mathbb{R}^N} \|w^h\|_1 + \chi_{Y^h}(y^h) + \chi_{C^h}(z^h) \quad & \text{s.t.} \quad w^h = \nabla^h u^h, y^h = H^h u^h, \\ & z^h = u^h, \end{aligned}$$

where  $Y^h := \{y^h \in \mathbb{R}^N : \frac{1}{N} \|S^h y^h - g^h\|_2^2 \leq \sigma^2\}$ .

The augmented Lagrangian of this problem is

$$\mathcal{L}(u^h, v^h, \lambda^h) = f(v^h) + \langle \lambda^h, B^h u^h - v^h \rangle + \frac{\beta}{2} \|B^h u^h - v^h\|_2^2,$$

with  $v^h = \begin{pmatrix} w^h \\ y^h \\ z^h \end{pmatrix} \in \mathbb{R}^{4N}$ ,  $f(v^h) = \|w^h\|_1 + \chi_{Y^h}(y^h) + \chi_{C^h}(z^h)$ ,  $B^h = \begin{pmatrix} \nabla^h \\ H^h \\ D^h(e_N) \end{pmatrix} \in \mathbb{R}^{4N \times N}$ ,

and  $\beta > 0$  denoting the penalty parameter. Hence the ADM for solving (9.27) runs as follows:

**Box-constrained ADMM:** Initialize  $v_0^h \in \mathbb{R}^{4N}$ ,  $\lambda_0^h \in \mathbb{R}^{4N}$  and set  $n = 0$ ;

- 1) Compute  $u_{n+1}^h \in \arg \min_{u^h} \langle \lambda_n^h, B^h u^h - v_n^h \rangle + \frac{\beta}{2} \|B^h u^h - v_n^h\|_2^2$
- 2) Compute  $v_{n+1}^h = \arg \min_{v^h} f(v^h) + \langle \lambda_n^h, B^h u_{n+1}^h - v^h \rangle + \frac{\beta}{2} \|B^h u_{n+1}^h - v^h\|_2^2$
- 3) Update  $\lambda_{n+1}^h = \lambda_n^h + \beta(B^h u_{n+1}^h - v_{n+1}^h)$
- 4) Stop or set  $n = n + 1$  and continue with step 1).

In order to obtain  $u_{n+1}^h$  in step 1) a linear system that may be diagonalized by the DFT is to solve. The solution of the minimization problem in step 2) might be computed as described in [57, Section 4.2]. More precisely, we have

$$\begin{aligned} v_{n+1}^h &= \arg \min_{v^h} f(v^h) + \langle \lambda_n^h, B^h u_{n+1}^h - v^h \rangle + \frac{\beta}{2} \|B^h u_{n+1}^h - v^h\|_2^2 \\ &= \arg \min_{v^h} f(v^h) + \frac{\beta}{2} \|v^h - (B^h u_{n+1}^h + \frac{\lambda_n^h}{\beta})\|_2^2 =: \text{prox}_{f/\beta} \left( B^h u_{n+1}^h + \frac{\lambda_n^h}{\beta} \right), \end{aligned}$$

where  $\text{prox}_f$  is called proximal operator of  $f$ . If we write  $v_n^h = u_{n+1}^h + \frac{\lambda_n^h}{\beta}$ , we can decompose  $\text{prox}_{f/\beta}(\cdot)$  as

$$\text{prox}_{f/\beta} \left( \begin{pmatrix} w^h \\ y^h \\ z^h \end{pmatrix} \right) = \left( \begin{array}{c} \text{prox}_{\|\cdot\|_1/\beta}(w^h) \\ \text{prox}_{\chi_{Y^h}/\beta}(y^h) \\ \text{prox}_{\chi_{C^h}/\beta}(z^h) \end{array} \right).$$

$\sigma$	$\alpha$	box-constrained pdN				box-constrained ADMM			
		PSNR	MSSIM	time	It	PSNR	MSSIM	time	It
0.3	0.34586	22.485	0.18619	18.844	41	22.2	0.17829	897.82	127
0.2	0.21408	24.054	0.24497	14.847	33	23.852	0.23917	808.67	100
0.1	0.096108	27.132	0.35201	14.91	33	26.963	0.35118	716.37	70
0.05	0.043393	30.567	0.47437	22.902	51	30.488	0.47503	656.66	48
0.01	0.0071847	40.417	0.75235	59.533	133	40.542	0.75864	454.45	24
0.005	0.0032996	45.164	0.8686	89.674	199	45.423	0.87718	501.59	24
$\emptyset$		31.6363	0.47975	36.785	81.667	31.5781	0.47991	672.5961	65.5

**Table 9.5:** Reconstruction of the cameraman-image corrupted by Gaussian white noise with standard deviation  $\sigma$ .

From [57] we know, that

$$\text{prox}_{\|\cdot\|_1/\beta}(w^h) = \begin{cases} w^h & \text{if } [|w^h|] = 0, \\ w^h - \min\{\frac{1}{\beta}, [|w^h|]\} \frac{w^h}{|w^h|} & \text{otherwise,} \end{cases}$$

and  $\text{prox}_{\chi_{Y^h}/\beta}(y^h)$  is a projection onto a weighted  $\ell^2$ -ball, which might be implemented as described in [65]. From the definition of the proximal operator we see that

$$\text{prox}_{\chi_{C^h}/\beta}(z^h) = \arg \min_{\tilde{z}^h \in C^h} \|\tilde{z}^h - z^h\|$$

is just the simple orthogonal projection of  $z^h$  onto  $C^h$ .

We recall that the ADMM converges for any  $\beta > 0$ , see for example [12, 36, 39]. In our numerical experiments we set  $\beta = 100$  and we use the same stopping criterion as suggested in [57].

### 9.6.3.2. pdN Versus ADMM

We start by comparing the performance of the proposed primal-dual semismooth Newton method (pdN) and the ADMM. In these experiments we assume that we know the optimal parameters  $\alpha$ , which are then used in the pdN. Note, that a fair comparison of these two methods is difficult, since they are solving different optimization problems, as already mentioned above. However, we still compare them in order to understand better the performance of the algorithms in the sequel section.

The comparison is performed for image denoising and image deblurring and the respective findings are collected in Table 9.5 and Table 9.6. From there we clearly observe, that the proposed pdN with  $\eta = 10^6$  reaches in all experiments the desired reconstruction significantly faster than the box-constrained ADMM. While the number of iterations for image denoising is approximately the same for both methods, for image deblurring the box-constrained pdN needs significantly less iterations than the other method. In particular, the pdN needs nearly the same amount of iterations independently of the application. However, more iterations for small  $\sigma$  are needed. Note, that the pdN converges at a superlinear rate and hence a faster convergence than the box-constrained ADMM is not surprising but supports the theory.

### 9.6.3.3. Image Denoising

In Table 9.7 and Table 9.8 we summarize our findings for image denoising. We observe that adding a box-constraint to the considered optimization problem leads to a possibly slight

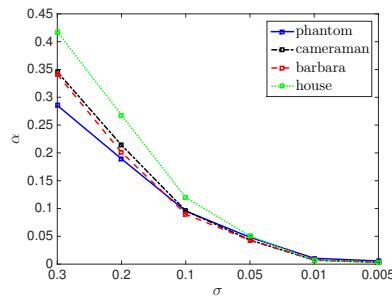
## II. Parameter Selection Methods for Total Variation Models

$\sigma$	$\alpha$	box-constrained pdN				box-constrained ADMM			
		PSNR	MSSIM	time	It	PSNR	MSSIM	time	It
0.3	0.2342	20.382	0.10678	1551.5	55	20.361	0.099691	2217.3	256
0.2	0.13169	20.981	0.13262	1434	41	20.978	0.12702	2593.1	265
0.1	0.051871	21.814	0.18113	1407.4	37	21.825	0.17536	3404.5	292
0.05	0.01951	22.484	0.22905	2691.4	51	22.501	0.22423	4065.4	305
0.01	0.0012674	24.293	0.34618	2440.5	126	24.237	0.34082	7185.3	358
0.005	0.00031869	25.451	0.40405	1903.4	149	25.377	0.3987	26985	1081
$\emptyset$		22.5674	0.2333	1904.6943	76.5	22.5464	0.22764	7741.7188	426.17

**Table 9.6:** Reconstruction of the cameraman-image corrupted by Gaussian blur and Gaussian white noise with standard deviation  $\sigma$ .

improvement in PSNR and MSSIM. While in some cases there is some improvement (see for example the image “numbers”) in other examples no improvement is gained (see for example the image “barbara”). In order to make the overall improvement more visible, in the last row of Table 9.7 and Table 9.8 we add the average PSNR and MSSIM of all computed restorations. It shows that in average we may expect a gain of around 0.05 PSNR and around 0.001 MSSIM, which is nearly nothing. Moreover, we observe, that the pAPS-algorithm computes the optimal  $\alpha$  for the box-constrained problem in average faster than the one for the non-box-constrained problem. We remark, that the box-constrained version needs less (or at maximum the same amount of) iterations as the version with  $\eta = 0$ . The reason for this might be that in each iterations, due to the thresholding of the approximation by the box-constraint, a longer or better step towards the minimizer than by the non-box-constrained pAPS-algorithm is performed. At the same time also the reconstructions of the box-constrained pAPS-algorithm yield higher PSNR and MSSIM than the ones obtained by the pAPS-algorithm with  $\eta = 0$ . The situation seems to be different for the ADMM. In average the box-constrained ADMM and the (non-box-constrained) ADMM need approximately the same run-time.

For several examples (i.e., the images “phantom”, “cameraman”, “barbara”, “house”) the choice of the regularization parameter by the box-constrained pAPS-algorithm with respect to the noise-level is depicted in Figure 9.4. Clearly, the parameter is selected to be smaller the less noise is present in the image.



**Figure 9.4:** Regularization parameter versus noise-level for the box-constrained pAPS in image denoising.

We are now wondering whether a box-constraint is more important when the regularization parameter is nonscalar, i.e., when  $\alpha : \Omega \rightarrow \mathbb{R}^+$  is a function. For computing suitable locally

## 9. Box-Constraint Total Variation

Image	$\sigma$	PSNR	MSSIM	pAPS with pdN time	$\eta = 0$	it	PSNR	MSSIM	pAPS with box-constrained pdN time	$\alpha$	it
phantom	0.3	19.274	0.30365	693.59	0.28721	40	19.302	0.30456	486.9	0.28544	20
	0.2	22.024	0.37461	553.44	0.19055	37	22.06	0.37521	393.67	0.18921	17
	0.1	27.471	0.44124	565.80	0.09621	37	27.518	0.4412	389.37	0.095365	17
	0.06	33.173	0.46878	888.33	0.048744	37	33.228	0.46847	421.18	0.04827	16
	0.01	46.809	0.50613	1605.4	0.010597	35	46.46	0.50559	791.34	0.010509	17
	0.005	51.055	0.52587	2245.3	0.0057362	35	51.846	0.52522	925.46	0.0056958	15
cameraman	0.3	22.485	0.18619	1591	0.34586	102	22.485	0.18621	847.53	0.34579	50
	0.2	24.054	0.24497	919.63	0.21408	66	24.056	0.24508	528.85	0.21398	32
	0.1	27.132	0.35201	580.91	0.096108	39	27.135	0.35221	646.44	0.096029	41
	0.06	30.567	0.47437	549.23	0.043593	24	30.571	0.47458	561.43	0.04336	25
	0.01	40.417	0.75235	677.79	0.0071847	12	40.418	0.75238	645.32	0.0071837	12
	0.005	45.164	0.86868	745.67	0.0032996	9	45.164	0.86861	701.1	0.0032995	9
barbara	0.3	20.618	0.30666	1419.7	0.34118	84	20.618	0.30666	735.83	0.34118	39
	0.2	21.649	0.39964	788.49	0.20109	47	21.649	0.39971	358.6	0.20105	18
	0.1	24.241	0.58555	336.17	0.089758	23	24.241	0.58564	319.09	0.089746	23
	0.06	27.884	0.75178	326.59	0.04286	15	27.885	0.7518	286.61	0.042858	16
	0.01	38.781	0.9322	345.64	0.0079133	9	38.781	0.9322	311.25	0.0079133	9
	0.005	44.056	0.9681	395.8	0.0037205	7	44.056	0.9681	330.97	0.0037205	7
house	0.3	23.827	0.1839	2750	0.41771	154	23.829	0.18392	1185.9	0.41763	75
	0.2	25.611	0.23397	1460.6	0.26795	116	25.611	0.23397	610.2	0.26796	45
	0.1	28.855	0.31916	690.33	0.11975	75	28.855	0.31916	348.92	0.11979	34
	0.06	32.094	0.4074	574.34	0.05059	39	32.094	0.40741	563.92	0.050502	40
	0.01	40.292	0.75174	493.84	0.0071206	11	40.292	0.75174	468.11	0.0071206	11
	0.005	44.989	0.86648	527	0.0033035	8	44.989	0.86648	302.38	0.0033035	8
lena	0.3	21.905	0.29155	1925	0.41731	120	21.905	0.29155	953.83	0.41731	57
	0.2	23.506	0.36317	1064.9	0.25937	85	23.506	0.36317	585.24	0.25937	39
	0.1	26.37	0.49351	537.69	0.11246	46	26.369	0.49351	302.32	0.11246	21
	0.06	29.615	0.622313	566.71	0.047771	27	29.615	0.62313	579.3	0.04777	28
	0.01	39.261	0.91371	526.89	0.0068096	97	39.262	0.91371	546.75	0.0068091	10
	0.005	44.672	0.97133	626.47	0.0032764	8	44.673	0.97133	497.97	0.0032762	8
bones	0.3	25.744	0.34395	5830.2	0.86048	310	25.743	0.34395	749.26	0.8605	35
	0.2	27.637	0.39821	2949	0.56086	238	27.637	0.39821	747.28	0.56086	45
	0.1	30.908	0.49398	122.8	0.27216	141	30.908	0.49398	364.71	0.27216	32
	0.06	34.284	0.58336	612.74	0.12735	87	34.284	0.58386	362.62	0.12735	41
	0.01	43.174	0.7449	386.03	0.020815	33	43.174	0.7449	479.77	0.020814	33
	0.005	47.493	0.80124	340.02	0.0091423	23	47.493	0.80124	470.83	0.0091423	23
cookies	0.3	21.466	0.31394	1117.4	0.38254	87	21.466	0.31396	857.99	0.38252	42
	0.2	23.136	0.40787	709.8	0.25117	62	23.136	0.40787	320.84	0.25118	12
	0.1	26.498	0.55614	398.59	0.12598	42	26.498	0.55614	290.67	0.12598	18
	0.06	30.292	0.67741	382.29	0.06257	30	30.293	0.67741	523.47	0.06257	30
	0.01	40.482	0.85926	414.09	0.011324	15	40.482	0.85926	567.65	0.011324	15
	0.005	45.39	0.91128	470.54	0.0052653	12	45.39	0.91128	653.55	0.0052653	12
numbers	0.3	17.593	0.33171	520.53	0.27654	31	17.862	0.35167	358.5	0.26337	6
	0.2	20.658	0.39035	415.03	0.18442	30	20.936	0.40758	375.14	0.17526	9
	0.1	26.259	0.44549	365.01	0.092576	30	26.56	0.4525	318.58	0.087618	8
	0.06	32.061	0.47618	466.24	0.046674	30	32.382	0.48698	321.63	0.044044	8
	0.01	45.511	0.51733	1039	0.009524	31	45.869	0.52176	467.07	0.0093273	7
	0.005	51.036	0.53022	1474.9	0.0053171	30	51.44	0.53149	463.78	0.0049401	5
<hr/>											
<hr/>											
<hr/>											
<hr/>											
<hr/>											
<hr/>											

**Table 9.7:** PSNR- and MSSIM-values of the reconstruction of different images corrupted by Gaussian white noise via pAPS using the primal-dual Newton method.

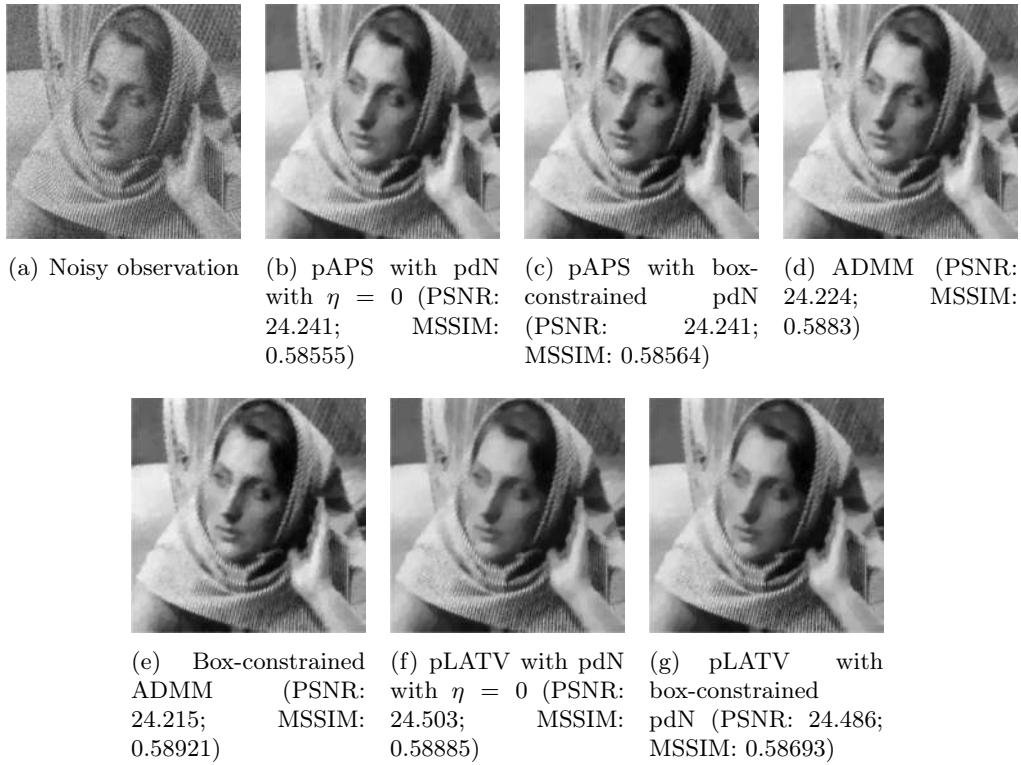
## II. Parameter Selection Methods for Total Variation Models

Image	$\sigma$	ADMM				box-constrained ADMM			
		PSNR	MSSIM	time	it	PSNR	MSSIM	time	it
phantom	0.3	18.956	0.29789	1245	118	18.922	0.29225	999.39	137
	0.2	21.615	0.37299	1203.8	108	21.538	0.36632	1068.9	123
	0.1	27.021	0.45439	868.12	79	26.941	0.44525	995.99	91
	0.05	32.823	0.5007	576.35	47	32.731	0.4847	650.9	53
	0.01	46.446	0.60023	444.34	23	47.36	0.54521	524.88	27
	0.005	53.734	0.5907	566.53	24	53.59	0.51937	913.28	38
cameraman	0.3	22.33	0.18003	889.65	99	22.2	0.17829	897.82	127
	0.2	23.895	0.23961	836.26	80	23.852	0.23917	808.67	100
	0.1	27.002	0.35006	690.45	60	26.963	0.35118	716.37	70
	0.05	30.513	0.47359	564.73	43	30.488	0.47503	656.66	48
	0.01	40.486	0.75493	409.62	22	40.542	0.75864	454.45	24
	0.005	45.329	0.8728	494.94	23	45.423	0.87718	501.59	24
barbara	0.3	20.6	0.30878	891.68	94	20.604	0.31256	683.92	112
	0.2	21.655	0.40225	827.72	74	21.654	0.40468	628.45	91
	0.1	24.224	0.5883	583.96	50	24.215	0.58921	547.54	61
	0.05	27.889	0.75319	449.51	35	27.874	0.75405	470.8	41
	0.01	38.921	0.9338	441.87	21	38.978	0.9341	374.69	24
	0.005	44.413	0.97035	503.68	21	44.584	0.97166	425.92	23
house	0.3	23.761	0.19254	1072.7	108	23.689	0.19468	674.76	128
	0.2	25.659	0.24287	949.97	85	25.609	0.24367	709.26	108
	0.1	28.907	0.32034	678	54	28.875	0.32115	596.95	70
	0.05	32.054	0.40609	532.99	35	32.029	0.40732	478.67	42
	0.01	40.394	0.7555	422.07	19	40.438	0.75891	331.36	21
	0.005	45.201	0.87241	492.11	20	45.278	0.8758	408.17	22
lena	0.3	21.834	0.29247	950.81	102	21.85	0.2995	570.59	115
	0.2	23.455	0.36437	724	81	23.46	0.36828	670.96	97
	0.1	26.349	0.49495	584.08	55	26.333	0.49605	641.47	67
	0.05	29.604	0.62425	456.63	39	29.588	0.62541	554.51	44
	0.01	39.422	0.91694	377.76	22	39.486	0.91822	438.7	25
	0.005	45.021	0.97391	470.81	24	45.106	0.97436	517.97	26
bones	0.3	25.829	0.35686	750.12	110	25.051	0.36748	611.23	115
	0.2	27.689	0.40363	734.13	90	27.486	0.41281	652.2	94
	0.1	30.942	0.48823	624.02	58	31.02	0.49267	755.06	75
	0.05	34.319	0.57951	343.83	35	34.37	0.57521	624.7	49
	0.01	43.011	0.73032	193.87	14	42.9	0.72002	270.92	16
	0.005	47.446	0.79549	204.98	12	47.432	0.78519	285.72	14
cookies	0.3	21.436	0.31503	789.78	102	21.461	0.32264	788.24	118
	0.2	23.129	0.40742	713.25	82	23.128	0.41261	772.93	99
	0.1	26.527	0.55478	554.95	55	26.506	0.55871	659.5	70
	0.05	30.364	0.67447	422.84	37	30.338	0.67746	589.77	45
	0.01	40.513	0.85593	234.35	15	40.589	0.85815	316.64	19
	0.005	45.382	0.90918	255.79	14	45.307	0.90892	319.82	16
numbers	0.3	17.257	0.32157	1640.7	147	17.487	0.34599	1311.9	146
	0.2	20.279	0.38382	1594	133	20.49	0.42	1215.8	134
	0.1	25.901	0.44284	981.06	95	26.125	0.48656	874.87	85
	0.05	31.847	0.47604	719.72	52	32.412	0.51241	532.91	46
	0.01	45.726	0.52365	643.88	26	47.089	0.51102	1001.3	56
	0.005	52.512	0.53468	641.43	29	52.859	0.51942	1756.9	83
$\emptyset$		32.0754	0.5386	671.7274	57.7292	32.1302	0.5389	671.9570	67.8958

**Table 9.8:** PSNR- and MSSIM-values of the reconstruction of different images corrupted by Gaussian white noise via the ADMM by solving the constrained versions.

varying  $\alpha$  we use the pLATV-algorithm proposed in [48], whereby we set in all considered examples the initial (nonscalar) regularization parameter to be constant  $10^{-2}$ . Note, that the continuity assumption on  $\alpha$  in (9.5) and (9.6) is not needed in our discrete setting, since  $\sum_{x \in \Omega^h} \alpha(x) |\nabla^h u^h(x)|$  is well defined for any  $\alpha \in \mathbb{R}^N$ . We approximate such  $\alpha$  for the problem (9.21) with  $\eta = 0$  (unconstrained) and with  $\eta = 10^6$  (box-constrained) and obtain also here that the gain with respect to PSNR and MSSIM is of the same order as in the scalar case, see Table 9.9.

For  $\sigma = 0.1$  and the image ‘‘barbara’’ we show in Figure 9.5 the reconstructions generated by the considered algorithms. As indicated by the quality measures, all the reconstructions look nearly alike, whereby in the reconstructions produced by the pLATV-algorithm details, like the pattern of the scarf, are (slightly) better preserved. The spatially varying  $\alpha$  of the pLATV-algorithm is depicted in Figure 9.6. There we clearly see, that at the scarf around the neck and shoulder the values of  $\alpha$  are small, allowing to preserve the details better.



**Figure 9.5:** Reconstruction from blurry and noisy data.

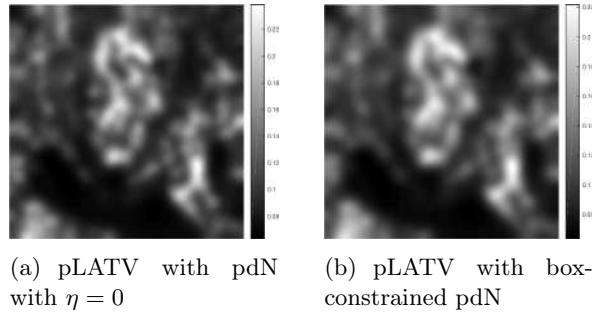
#### 9.6.3.4. Image Deblurring

Now we consider the images in Figure 9.1(a)-(c), convolve them first with a Gaussian kernel of size  $9 \times 9$  and standard deviation 3 and then add some Gaussian noise with mean 0 and standard deviation  $\sigma$ . Here we again compare the results obtained by the pAPS-algorithm, the ADMM, and the pLATV-algorithm for the box-constrained and non-box-constrained problems. Our findings are summarized in Table 9.10. Also here we observe a slight improvement with a box-

## II. Parameter Selection Methods for Total Variation Models

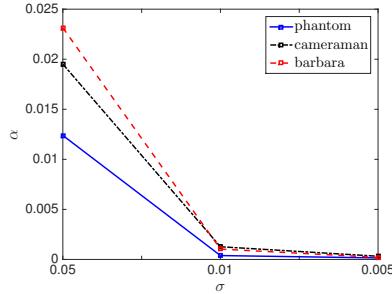
Image	$\sigma$	pLATV with pdN with $\eta = 0$				pLATV with box-constrained pdN			
		PSNR	MSSIM	time	it	PSNR	MSSIM	time	it
phantom	0.3	19.629	0.31986	517.78	13	19.744	0.32143	710.59	16
	0.2	22.405	0.38578	286.66	13	22.525	0.38706	607.27	15
	0.1	27.802	0.44701	389.34	13	27.936	0.44762	552.49	14
	0.05	33.48	0.47301	526.89	13	33.57	0.4733	571.06	14
	0.01	46.546	0.50749	139.72	4	46.625	0.50646	98.973	3
	0.005	52.038	0.52569	112.1	3	52.062	0.52512	105.89	3
cameraman	0.3	22.382	0.18185	775.5	15	22.393	0.18186	838.55	19
	0.2	24.032	0.24038	605.56	13	24.029	0.23937	593.9	16
	0.1	27.16	0.35301	304.83	12	27.175	0.35285	415.09	13
	0.05	30.702	0.4745	310.03	10	30.696	0.47315	262.59	10
	0.01	40.647	0.73386	135.88	4	40.647	0.73386	144.01	4
	0.005	45.292	0.86678	365.19	7	45.292	0.86678	357.42	7
barbara	0.3	20.527	0.30278	572.44	16	20.516	0.3023	652.83	18
	0.2	21.73	0.39872	411.5	13	21.729	0.39882	459.09	14
	0.1	24.503	0.58885	285.8	9	24.486	0.58693	302.46	10
	0.05	28.196	0.75135	309.53	7	28.198	0.75151	304.89	7
	0.01	38.898	0.92794	424.8	18	38.898	0.92794	425.12	18
	0.005	44.186	0.96871	147.57	5	44.186	0.96871	150.91	5
house	0.3	23.661	0.18529	593.22	18	23.704	0.18526	698.35	22
	0.2	25.507	0.23789	478.33	17	25.51	0.23741	522.86	18
	0.1	28.736	0.32695	304.43	13	28.741	0.32581	332.47	14
	0.05	31.94	0.4217	184.75	11	31.943	0.42182	320.22	11
	0.01	40.423	0.73752	451.47	13	40.423	0.73752	377.89	13
	0.005	45.118	0.85458	286.48	7	45.118	0.85458	222.54	7
lena	0.3	21.829	0.29245	624.96	19	21.828	0.29232	727.69	21
	0.2	23.442	0.3634	471.91	16	23.445	0.36326	538.81	17
	0.1	26.403	0.49438	347.04	13	26.406	0.49419	374.1	14
	0.05	29.703	0.62778	396.43	8	29.704	0.62784	403.49	8
	0.01	39.324	0.91256	652.2	16	39.324	0.91256	659.92	16
	0.005	44.736	0.97081	244.61	5	44.737	0.97081	237.28	5
bones	0.3	25.633	0.36194	869.22	37	25.518	0.35855	1122.8	38
	0.2	27.6	0.41895	665.85	33	27.506	0.41674	823.95	33
	0.1	30.964	0.51162	419.72	26	30.862	0.51085	435.6	24
	0.05	34.413	0.59991	291.13	19	34.408	0.59947	374.07	19
	0.01	43.521	0.76261	98.45	6	43.521	0.76261	93.765	6
	0.005	47.564	0.7997	103.29	6	47.564	0.7997	95.637	6
cookies	0.3	21.415	0.31722	597.85	17	21.371	0.3164	653.83	18
	0.2	23.126	0.40895	500.29	16	23.107	0.4087	485.2	15
	0.1	26.487	0.55758	237.87	12	26.501	0.55811	332.12	12
	0.05	30.301	0.68332	413.36	12	30.295	0.68292	175.11	9
	0.01	40.525	0.86052	261.76	8	40.525	0.86052	236.96	8
	0.005	45.499	0.90364	229.85	5	45.499	0.90364	174.58	5
numbers	0.3	17.627	0.33232	421.86	11	17.821	0.34957	654.22	20
	0.2	20.7	0.39072	313.32	10	20.888	0.40559	556.07	19
	0.1	26.296	0.4455	258.05	10	26.528	0.45766	440.13	17
	0.05	32.104	0.47624	339.03	10	32.369	0.48578	501.47	15
	0.01	45.527	0.51735	92.644	2	45.938	0.52169	59.513	2
	0.005	51.265	0.53078	207.86	5	51.536	0.53125	257.45	6
$\emptyset$		32.1155	0.5365	374.5490	12.2708	32.1531	0.5375	425.8590	13.4167

**Table 9.9:** PSNR- and MSSIM-values of the reconstruction of different images corrupted by Gaussian white noise via pLATV using the primal-dual Newton method.



**Figure 9.6:** Spatially varying regularization parameter generated by the respective pLATV-algorithm.

constraint with respect to PSNR and MSSIM. The choice of the regularization parameters by the box-constrained pAPS-algorithm is depicted in Figure 9.7. In Figure 9.8 we present for the image ‘‘cameraman’’ and  $\sigma = 0.01$  the reconstructions produced by the respective methods. Also here, as indicated by the quality measures, all the restorations look nearly the same. The locally varying  $\alpha$  generated by the pLATV-algorithm are depicted in Figure 9.9.



**Figure 9.7:** Regularization parameter versus noise-level for the box-constrained pAPS in image deblurring.

#### 9.6.3.5. Image Inpainting

The problem of filling in and recovering missing parts in an image is called *image inpainting*. We call the missing parts *inpainting domain* and denote it by  $D \subset \Omega$ . The linear bounded operator  $K$  is then a multiplier, i.e.,  $Ku = 1_{\Omega \setminus D} \cdot u$ , where  $1_{\Omega \setminus D}$  is the indicator function of  $\Omega \setminus D$ . Note, that  $K$  is not injective and hence  $K^*K$  is not invertible. Hence in this experiment we need to set  $\mu > 0$  so that we can use the proposed primal-dual semismooth Newton method. In particular, as mentioned above, we choose  $\mu = 10^{-6}$ .

In the considered experiments the inpainting domain are gray bars as shown in Figure 9.10(a), where additionally additive white Gaussian noise with  $\sigma = 0.1$  is present. In particular we consider examples with  $\sigma \in \{0.3, 0.2, 0.1, 0.05, 0.01, 0.005\}$ . The performance of the pAPS- and pLATV-algorithm with and without a box-constraint reconstructing the considered examples are summarized in Table 9.11 and Table 9.12. We observe, that adding a box-constraint does not seem to change the restoration considerably. However, as in the case

## II. Parameter Selection Methods for Total Variation Models

Image	$\sigma$	pAPS with pdN with $\eta = 0$				pAPS with box-constrained pdN			
		PSNR	MSSIM	time	it	PSNR	MSSIM	time	it
phantom	0.05	16.41	0.21695	11603	11	16.801	0.23085	33002	11
	0.01	18.624	0.30997	12663	10	18.861	0.32574	18744	8
	0.005	20.554	0.36713	5892.9	6	21.01	0.38225	17245	9
cameraman	0.05	22.482	0.2288	8328.5	11	22.484	0.22905	24162	11
	0.01	24.281	0.34554	4799.8	5	24.293	0.34618	13102	5
	0.005	25.436	0.4035	5611.4	8	25.451	0.40405	12033	7
barbara	0.05	21.363	0.37833	5422.8	8	21.363	0.37832	10657	8
	0.01	22.052	0.49312	2820.6	5	22.052	0.49311	6188.1	5
	0.005	23.084	0.57223	4275.5	13	23.084	0.57223	11121	13
$\emptyset$		21.5874	0.3684	6824	8.5556	21.7108	0.37353	16250	8.5556

Image	$\sigma$	ADMM				box-constrained ADMM			
		PSNR	MSSIM	time	it	PSNR	MSSIM	time	it
phantom	0.05	16.427	0.2691	6243.1	465	16.811	0.27613	8186.6	581
	0.01	18.581	0.3094	34683	1424	18.809	0.32544	33141	1327
	0.005	20.505	0.3677	40801	1426	20.906	0.38285	43753	1485
cameraman	0.05	22.51	0.2242	2870.1	214	22.501	0.22423	4065.4	305
	0.01	24.236	0.34073	8268.2	428	24.237	0.34082	7185.3	358
	0.005	25.373	0.3986	30944	1200	25.377	0.3987	26985	1081
barbara	0.05	21.417	0.38039	1996.6	169	21.421	0.38113	2807.4	237
	0.01	22.036	0.48965	13282	678	22.038	0.48964	12012	627
	0.005	23.043	0.56871	40949	1527	23.038	0.56841	37430	1423
$\emptyset$		21.5699	0.37205	20004	836.78	21.6821	0.37637	19507	824.89

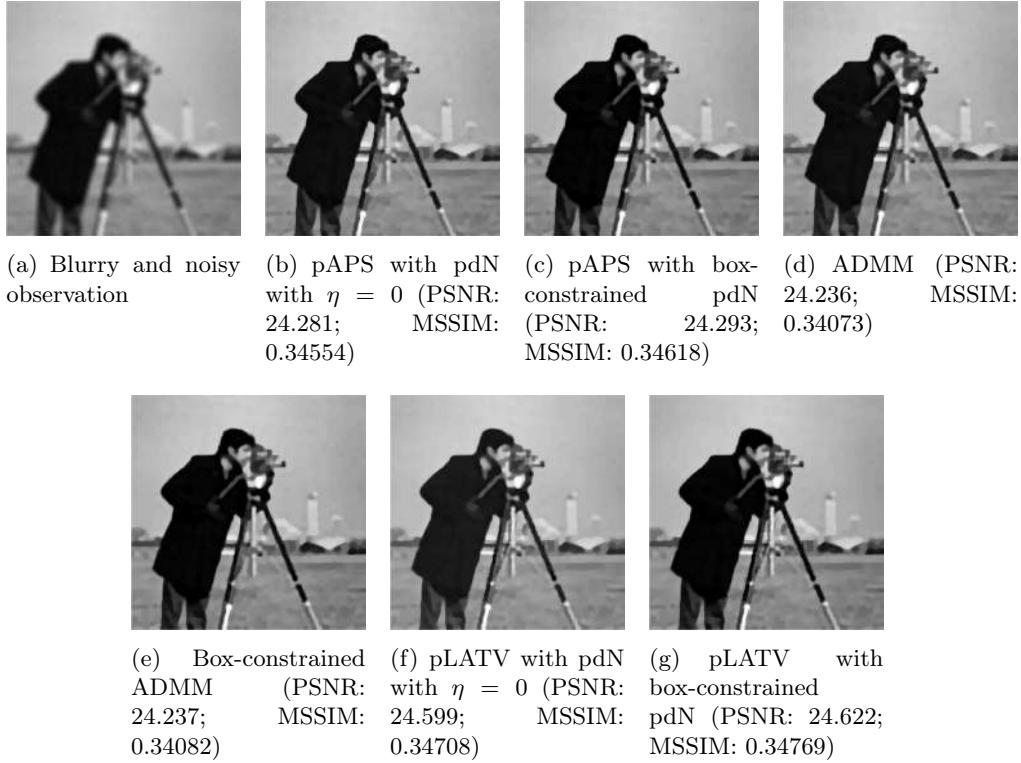
Image	$\sigma$	pLATV with pdN with $\eta = 0$				pLATV with box-constrained pdN			
		PSNR	MSSIM	time	it	PSNR	MSSIM	time	it
phantom	0.05	16.43	0.21966	14732	12	16.824	0.23305	29567	10
	0.01	18.967	0.31788	39014	49	19.602	0.33679	11950	52
	0.005	20.729	0.36861	49658	67	22.041	0.39052	11940	71
cameraman	0.05	22.533	0.22964	26380	22	22.534	0.22973	70208	22
	0.01	24.599	0.34708	57513	47	24.622	0.34769	113175	47
	0.005	25.77	0.40389	39662	60	25.796	0.40451	110646	60
barbara	0.05	21.377	0.3808	26779	24	21.377	0.3808	49808	24
	0.01	22.432	0.50565	22433	51	22.448	0.50625	78720	52
	0.005	23.652	0.59532	16168	61	23.658	0.59545	53559	61
$\emptyset$		21.8319	0.37428	32482	43.667	22.1003	0.38053	81893	44.333

**Table 9.10:** PSNR- and MSSIM-values of the reconstruction of different images corrupted by Gaussian blur and Gaussian white noise via pAPS or pLATV using the primal-dual Newton method or via the ADMM by solving the constrained versions.

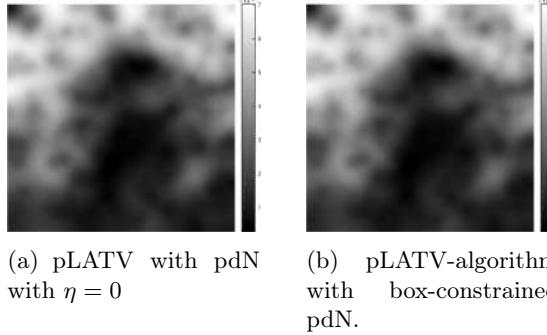
of image denoising, the pAPS-algorithm with box-constrained pdN needs less iterations and hence less time than the same algorithm without a box-constraint to reach the stopping criterion. Figure 9.10 shows a particular example for image inpainting and denoising with  $\sigma = 0.1$ . It demonstrates that visually there is nearly no difference between the restoration obtained by the considered approaches. Moreover, we observe that the pLATV-algorithm seems to be not suited to the task of image inpainting. A reason for this might be, that the pLATV-algorithm does not take the inpainting domain correctly into account. This is visible in Figure 9.11 where the spatially varying  $\alpha$  seems to be chosen small in the inpainting domain, which not necessarily seems to be a suitable choice.

### 9.6.3.6. Reconstruction from Partial Fourier-Data

In magnetic resonance imaging one wishes to reconstruct an image which is only given by partial Fourier data and additionally distorted by some additive Gaussian noise with zero



**Figure 9.8:** Reconstruction from blurry and noisy data.



**Figure 9.9:** Spatially varying regularization parameter generated by the respective pLATV-algorithm.

mean and standard deviation  $\sigma$ . Hence, the linear bounded operator is  $K = S \circ \mathcal{F}$ , where  $\mathcal{F}$  is the 2D Fourier matrix and  $S$  is a downsampling operator which selects only a few output frequencies. The frequencies are usually sampled along radial lines in the frequency domain, in particular in our experiments along 32 radial lines, as visualized in Figure 9.12.

In our experiments we consider the images of Figure 9.2, transformed to its Fourier frequencies. As already mentioned, we sample the frequencies along 32 radial lines and add some Gaussian noise with zero mean and standard deviation  $\sigma$ . In particular, we consider different

## II. Parameter Selection Methods for Total Variation Models

Image	$\sigma$	PSNR	pAPS with pdN with $\eta = 0$				pAPS with box-constrained pdN			
			MSSIM	time	$\alpha$	it	PSNR	MSSIM	time	$\alpha$
lena	0.3	21.151	0.26378	1709.9	0.37358	105	21.151	0.26378	958.59	0.37358
	0.2	22.555	0.33033	1075.8	0.23336	72	22.555	0.33032	578.35	0.23336
	0.1	24.922	0.44992	578.34	0.10369	41	24.922	0.44992	370.42	0.10369
	0.05	27.005	0.56734	513.1	0.044922	25	27.005	0.56735	507.37	0.044919
	0.01	29.618	0.82318	524.73	0.006614	9	29.618	0.82319	516.86	0.0066133
	0.005	29.912	0.87427	569.85	0.00319	8	29.912	0.87427	674.02	0.0031896
$\emptyset$		25.83	0.57403	617.22		36	25.83	0.57404	477.78	18.5

**Table 9.11:** PSNR- and MSSIM-values for the application inpainting via pAPS.

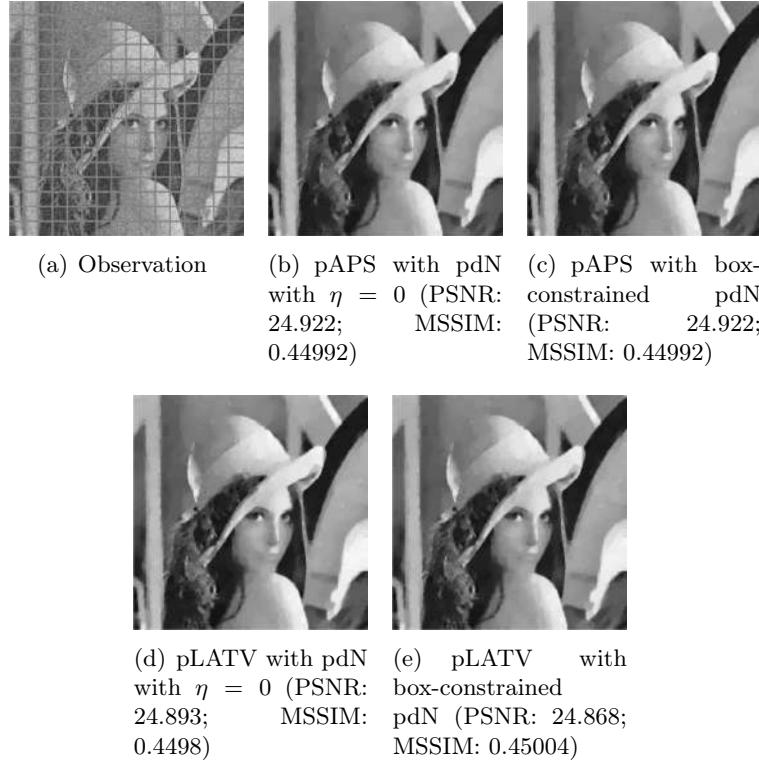
Image	$\sigma$	pLATV with pdN with $\eta = 0$				pLATV with box-constrained pdN			
		PSNR	MSSIM	time	it	PSNR	MSSIM	time	it
lena	0.3	21.027	0.26266	684.18	14	21.036	0.26239	836.18	16
	0.2	22.45	0.32986	555.66	11	22.457	0.32939	584.66	12
	0.1	24.893	0.4498	446.84	12	24.868	0.45004	400.3	9
	0.05	26.982	0.56904	474.61	9	26.983	0.56912	460.22	9
	0.01	29.621	0.82242	796.53	16	29.621	0.82242	775.88	16
	0.005	29.987	0.87461	285.24	5	29.987	0.87461	284.79	5
$\emptyset$		25.7643	0.56161	428.51	9.3333	25.765	0.56173	468.3961	9.6667

**Table 9.12:** PSNR- and MSSIM-values for the application inpainting via pLATV.

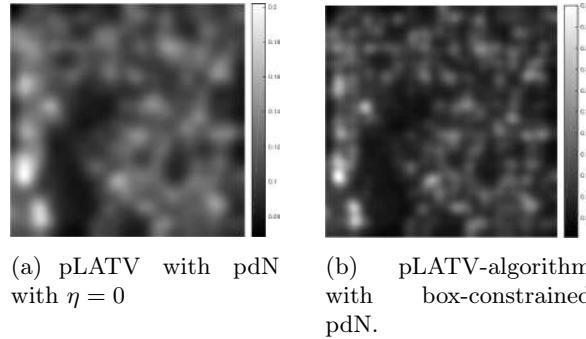
noise-levels, i.e.,  $\sigma = \{0.3, 0.2, 0.1, 0.05, 0.01, 0.005\}$ . We reconstruct the obtained data via the pAPS- and pLATV-algorithm by using the semismooth Newton method first with  $\eta = 0$  (no box-constraint) and then with  $\eta = 10^6$  (with box-constraint). In Table 9.13 we collect our findings. We observe that the pLATV-algorithm seems not to be suitable for this task, since it is generating inferior results. For scalar  $\alpha$  we observe as before, that a slight improvement with respect to PSNR and MSSIM is expectable when a box-constraint is used. In Figure 9.13 we present the reconstructions generated by the considered algorithms for a particular example, demonstrating the visual behavior of the methods.

### 9.6.3.7. Reconstruction from Sampled Radon-Transform

In computerized tomography instead of a Fourier-transform a Radon-transform is used in order to obtain a visual image from the measured physical data. Also here the data is obtained along radial lines. Here we consider the Shepp-Logan phantom, see Figure 9.14(a), and a slice of a body, see Figure 9.15(a). The sinogram in Figure 9.14(b) and Figure 9.15(b) are obtained by sampling along 30 and 60 radial lines, respectively. Note, that the sinogram is in general noisy. Here the data is corrupted by Gaussian white noise with standard deviation  $\sigma$ , whereby



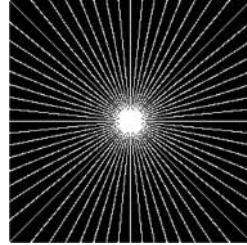
**Figure 9.10:** Simultaneous image inpainting and denoising with  $\sigma = 0.1$ .



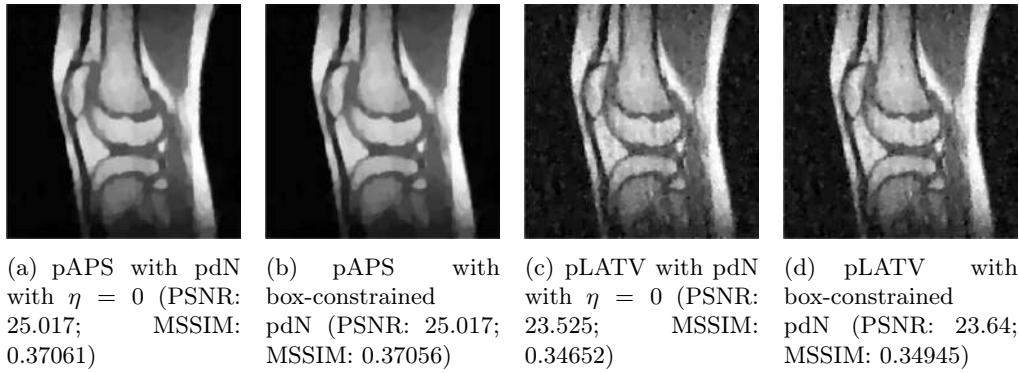
**Figure 9.11:** Spatially varying regularization parameter generated by the respective pLATV-algorithm.

$\sigma = 0.1$  for the data of the Shepp-Logan phantom and  $\sigma = 0.05$  for the data of the slice of the head. Using the inverse Radon-transform we obtain Figure 9.16(a) and 9.16(d), which is obviously a suboptimal reconstruction. A more sophisticated approach utilizes the  $L^2$ -TV model which yields the reconstruction depicted in Figure 9.16(b) and 9.16(e), where we use the pAPS-algorithm and the proposed primal-dual algorithm with  $\eta = 0$ . However, since an image can be assumed to have nonnegative values, we may incorporate a nonnegativity constraint via the box-constrained  $L^2$ -TV model yielding the result in Figure 9.16(c) and 9.16(f), which

## II. Parameter Selection Methods for Total Variation Models



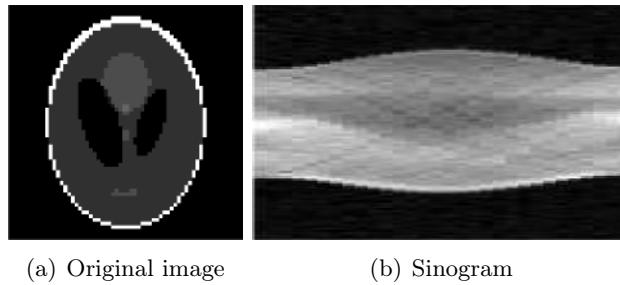
**Figure 9.12:** Sampling domain in the frequency plane, i.e., sampling operator  $S$ .



**Figure 9.13:** Reconstruction from sampled Fourier data.

is a much better reconstruction. Also here the parameter  $\alpha$  is automatically computed by the pAPS-algorithm and the nonnegativity constraint is incorporated by setting  $\eta = 10^6$  in the semismooth Newton method. In order to compute the Radon-matrix in our experiments we used the FlexBox [29].

Other applications where a box-constraint, and in particular a nonnegativity improves the image reconstruction quality significantly include for example magnetic particle imaging, see for example [62] and references therein.

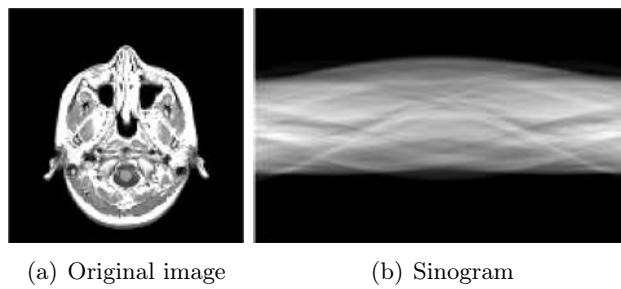


**Figure 9.14:** The Shepp-Logan phantom image of size  $64 \times 64$  pixels and its measured sinogram.

Image	$\sigma$	pAPS with pdN with $\eta = 0$			pAPS with box-constrained pdN		
		PSNR	MSSIM	CPU-time	PSNR	MSSIM	CPU-time
Shepp- Logan phantom	0.3	18.888	0.16233	3787.2	19.000	0.1685	5509.2
	0.2	20.524	0.21302	2844.9	20.696	0.22086	3673.5
	0.1	24.256	0.2905	1884.7	24.496	0.29896	2582.3
	0.05	28.639	0.34972	2008.5	28.948	0.35833	2115
	0.01	40.168	0.42734	1993.3	40.711	0.43325	1349.5
	0.005	45.263	0.44714	2225.4	45.933	0.45199	951.49
knee	0.3	21.606	0.26553	22466	21.606	0.26553	36054
	0.2	22.985	0.30965	15705	22.985	0.30965	31072
	0.1	25.017	0.37061	11561	25.017	0.37056	24994
	0.05	26.443	0.41652	8803.4	26.445	0.41661	21056
	0.01	27.912	0.47141	4996.9	27.959	0.47267	11707
	0.005	28.035	0.47683	6076.9	28.089	0.47843	13116
$\emptyset$		27.4781	0.35005	7029.4365	27.657	0.35378	12848.2064

Image	$\sigma$	pLATV with pdN with $\eta = 0$			pLATV with box-constrained pdN		
		PSNR	MSSIM	CPU-time	PSNR	MSSIM	CPU-time
Shepp- Logan phantom	0.3	18.99	0.16078	5445.3	17.148	0.11219	15500
	0.2	20.567	0.21006	3179.1	19.324	0.17395	11719
	0.1	24.376	0.29028	2491.5	23.51	0.27083	4623.8
	0.05	28.569	0.34645	1926.1	28.303	0.34392	7125.8
	0.01	39.475	0.41775	266.7	39.579	0.42053	695.74
	0.005	43.782	0.43085	465.09	43.627	0.43096	1373.9
knee	0.3	15.583	0.18089	17413	16.011	0.186	17750
	0.2	18.87	0.24419	11640	19.227	0.25069	14414
	0.1	23.525	0.34652	3663.5	23.64	0.34945	9220.4
	0.05	26.307	0.41393	1545.6	26.341	0.4159	4165.6
	0.01	27.044	0.46069	4091.1	27.055	0.4612	12059
	0.005	24.773	0.41841	10499	24.639	0.4172	34409
$\emptyset$		25.9885	0.3267	5218.8	25.7003	0.3194	11088

**Table 9.13:** PSNR- and MSSIM-values of the reconstruction of sampled Fourier data corrupted by Gaussian white noise via the pAPS- and pLATV-algorithm using the primal-dual Newton method.

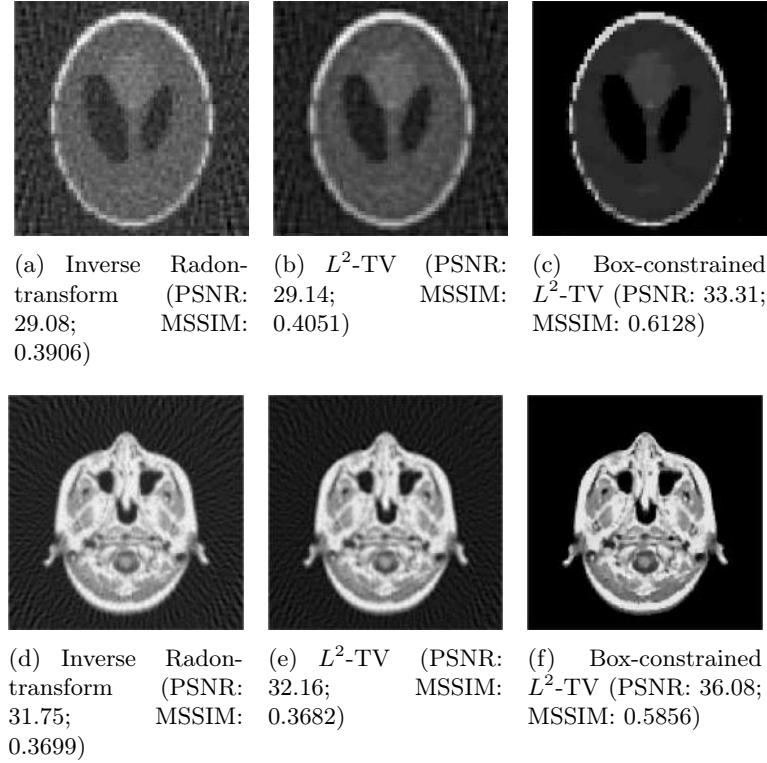


**Figure 9.15:** Slice of a human head and its measured sinogram.

## 9.7. Automated Parameter Selection

We recall, that if the noise-level  $\sigma$  is not known, then the problems (9.1) and (9.3) cannot be considered. Moreover, the selection of the parameter  $\alpha$  in (9.2) cannot be achieved by using the pAPS-algorithm, since this algorithm is based on (9.1). Note, that also other methods,

## II. Parameter Selection Methods for Total Variation Models



**Figure 9.16:** Reconstruction from noisy data.

like the unbiased predictive risk estimator method (UPRE) [51, 54] and approaches based on the Stein unbiased risk estimator method (SURE) [8, 33, 28, 35, 37] use knowledge of the noise-level and hence cannot be used for selecting a suitable parameter if  $\sigma$  is unknown.

If we assume that  $\sigma$  is unknown but the image intensity range of the original image  $\hat{u}$  is known, i.e.,  $\hat{u} \in [c_{\min}, c_{\max}]$ , then we may use this information for choosing the parameter  $\alpha$  in (9.2). This may be performed by applying the following algorithm:

**Box-constrained automatic parameter selection (bcAPS):** Initialize  $\alpha_0 > 0$  (sufficiently small) and set  $n := 0$

1. Solve  $u_n \in \arg \min_{u \in BV(\Omega)} \|Ku - g\|_{L^2(\Omega)}^2 + \alpha_n \int_{\Omega} |Du|$ .
2. If  $u_n \notin [c_{\min}, c_{\max}]$  increase  $\alpha_n$  (i.e.,  $\alpha_{n+1} := \tau \alpha_n$  with  $\tau > 1$ ), else STOP.
3. Set  $n := n + 1$  and continue with step 1.

Here  $\tau > 1$  is an arbitrary parameter chosen manually such that the generated restoration  $u$  is not over-smoothed, i.e., there exist  $x \in \Omega$  such that  $u(x) \approx c_{\min}$  and/or  $u(x) \approx c_{\max}$ . In our experiments it turned out that  $\tau = 1.05$  seems to be a reasonable choice, so that the generated solution has the wished property.

$\sigma$	bcAPS				pAPS			
	PSNR	MSSIM	time	$\alpha$	PSNR	MSSIM	time	$\alpha$
0.3	22.230	0.17478	1065.8	0.381058	22.485	0.18619	1591	0.34586
0.2	23.637	0.22552	1084.9	0.245634	24.054	0.24497	919.63	0.21408
0.1	26.621	0.32588	840.2	0.112528	27.132	0.35201	580.91	0.096108
0.05	29.388	0.41062	817.9	0.059676	30.567	0.47437	549.23	0.043393
0.01	39.332	0.70321	552.4	0.009346	40.417	0.75235	677.79	0.0071847
0.005	44.508	0.84591	415.6	0.003883	45.164	0.8686	745.67	0.0032996

**Table 9.14:** PSNR- and MSSIM-values of the reconstruction of the cameraman-image corrupted by Gaussian white noise with standard deviation  $\sigma$  via the bcAPS algorithm using the primal-dual Newton method with  $\eta = 0$ .

$\sigma$	bcAPS				pAPS			
	PSNR	MSSIM	time	$\alpha$	PSNR	MSSIM	time	$\alpha$
0.05	22.304	0.21524	76216.3	0.027598	22.482	0.2288	8328.5	0.019529
0.01	22.956	0.26523	86068.9	0.010401	24.281	0.34554	4799.8	0.0012714
0.005	23.024	0.27018	96837.0	0.009434	25.436	0.4035	5611.4	0.00031872

**Table 9.15:** PSNR- and MSSIM-values of the reconstruction of the cameraman-image corrupted Gaussian blur and Gaussian white noise with standard deviation  $\sigma$  via the bcAPS algorithm using the primal-dual Newton method with  $\eta = 0$ .

### 9.7.1. Numerical Examples

In our experiments the minimization problem in step 1 of the bcAPS algorithm is approximately solved by the proposed primal-dual semismooth Newton method with  $\eta = 0$ . We set the initial regularization parameter  $\alpha_0 = 10^{-4}$  for image denoising and  $\alpha_0 = 10^{-3}$  for image deblurring. Moreover, we set  $\tau = 1.05$  in the bcAPS-algorithm to increase the regularization parameter.

Experiments for image denoising, see Table 9.14, show that the bcAPS-algorithm finds suitable parameters in the sense that the PSNR and MSSIM of these reconstructions is similar to the ones obtained with the pAPS-algorithm (when  $\sigma$  is known); also compare with Table 9.7 and Table 9.10. This is explained by the observation that also the regularization parameters  $\alpha$  calculated by the bcAPS-algorithm do not differ much from the ones obtained via the pAPS-algorithm. For image deblurring, see Table 9.15, the situation is not so persuasive. In particular, the obtained regularization parameter of the two considered methods differ more significantly than before, resulting in different PSNR and MSSIM. However, in the case  $\sigma = 0.05$  the considered quality measures of the generated reconstructions are nearly the same.

We also remark, that in all the experiments the pAPS-algorithm generated reconstructions, which have larger PSNR and MSSIM than the ones obtained by the bcAPS-algorithm. From this observation it seems more useful to know the noise-level than the image intensity range. However, if the noise-level is unknown but the image intensity is known, then the bcAPS-algorithm may be a suitable choice.

## II. Parameter Selection Methods for Total Variation Models

### 9.8. Conclusion

In this work we investigated the quality of restored images when the image intensity range of the original image is additionally incorporated into the  $L^2$ -TV model as a box-constraint. We observe that this box-constraint may indeed improve the quality of reconstructions. However, if the observation already fulfills the box-constraint, then it clearly does not change the solution at all. Moreover, in a lot of applications the proper choice of the regularization parameter seems much more important than an additional box-constraint. Nevertheless, also then a box-constraint may improve the quality of the restored image, although the improvement is then only very little. On the contrary the additional box-constraint may improve the computational time significantly. In particular, for image deblurring and in magnetic resonance imaging using the pAPS-algorithm the computational time is about doubled, while the quality of the restoration is basically not improved. This suggests, that for these applications an additional box-constraint may not be reasonable. Note, that the run-time of the ADMM is independent whether a box-constraint is used or not.

For certain applications, as in computerized tomography, a box-constraint (in particular a nonnegativity constraint) improves the reconstruction considerably. Hence, the question rises under which conditions an additional box-constraint indeed has significant influence on the reconstruction when the present parameters are chosen in a nearly optimal way.

If the noise-level of an corrupted image is unknown but the image intensity range of the original image is at hand, then the image intensity range may be used to calculate a suitable regularization parameter  $\alpha$ . This can be done as explained in Section 9.7. Potential future research may consider different approaches, as for example in an optimal control setting. Then one may want to solve

$$\begin{aligned} & \min_{u,\alpha} \| \max\{u - c_{\max}, 0\} \|_{L^2(\Omega)}^2 + \| \max\{c_{\min} - u, 0\} \|_{L^2(\Omega)}^2 + \kappa J(\alpha) \\ \text{s.t. } & u \in \arg \min_u \| Ku - g \|_{L^2(\Omega)}^2 + \alpha \int_{\Omega} |Du|, \end{aligned}$$

where  $\kappa > 0$  and  $J$  is a suitable functional, cf. [27, 43, 47] for other optimal control approaches in image reconstruction.

### 9.9. References

- [1] M. Alkämper and A. Langer. Using DUNE-ACFem for non-smooth minimization of bounded variation functions. *Archive of Numerical Software*, 5(1):3–19, 2017.
- [2] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE Transactions on Signal Processing*, 45(4):913–917, 1997.
- [3] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, 2000.
- [4] H. Attouch, G. Buttazzo, and G. Michaille. *Variational Analysis in Sobolev and BV Spaces*. MOS-SIAM Series on Optimization. SIAM, Philadelphia, PA, 2014. Applications to PDEs and optimization.
- [5] J. M. Bardsley and C. R. Vogel. A nonnegatively constrained convex programming method for image reconstruction. *SIAM Journal on Scientific Computing*, 25(4):1326–1343, 2004.

## 9. Box-Constraint Total Variation

- [6] S. Bartels. *Numerical Methods for Nonlinear Partial Differential Equations*, volume 14. Springer, 2015.
- [7] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18(11):2419–2434, 2009.
- [8] T. Blu and F. Luisier. The SURE-LET approach to image denoising. *IEEE Transactions on Image Processing*, 16(11):2778–2786, 2007.
- [9] R. I. Bot and C. Hendrich. A Douglas–Rachford type primal-dual method for solving inclusions with mixtures of composite and parallel-sum type monotone operators. *SIAM Journal on Optimization*, 23(4):2541–2565, 2013.
- [10] R. I. Bot and C. Hendrich. Convergence analysis for a primal-dual monotone+ skew splitting algorithm with applications to total variation minimization. *Journal of Mathematical Imaging and Vision*, 49(3):551–568, 2014.
- [11] A. C. Bovik. *Handbook of Image and Video Processing*. Academic press, 2010.
- [12] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.
- [13] M. Burger, A. Sawatzky, and G. Steidl. First order algorithms in variational image processing. *arXiv preprint arXiv:1412.4237*, 2014.
- [14] V. C. Cao, J. C. D. L. Reyes, and C.-B. Schönlieb. Learning optimal spatially-dependent regularization parameters in total variation image restoration. *Inverse Problems*, 33(7):074005, 2017.
- [15] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97, 2004.
- [16] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, 1997.
- [17] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [18] R. H. Chan, C.-W. Ho, and M. Nikolova. Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. *IEEE Transactions on Image Processing*, 14(10):1479–1485, 2005.
- [19] R. H. Chan, M. Tao, and X. Yuan. Constrained total variation deblurring models and fast algorithms based on alternating direction method of multipliers. *SIAM Journal on Imaging Sciences*, 6(1):680–697, 2013.
- [20] T. F. Chan, G. H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.
- [21] P. L. Combettes and B. C. Vũ. Variable metric forward–backward splitting with applications to monotone inclusions in duality. *Optimization*, 63(9):1289–1318, 2014.
- [22] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.
- [23] L. Condat. A primal–dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *Journal of Optimization Theory and Applications*, 158(2):460–479, 2013.
- [24] J. Darbon and M. Sigelle. *A Fast and Exact Algorithm for Total Variation Minimization*, pages 351–359. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

## II. Parameter Selection Methods for Total Variation Models

- [25] J. Darbon and M. Sigelle. Image restoration with discrete constrained total variation. I. Fast and exact optimization. *Journal of Mathematical Imaging and Vision*, 26(3):261–276, 2006.
- [26] I. Daubechies, G. Teschke, and L. Vese. Iteratively solving linear inverse problems under general convex constraints. *Inverse Problems and Imaging*, 1(1):29–46, 2007.
- [27] J. C. De los Reyes and C.-B. Schönlieb. Image denoising: Learning the noise model via nonsmooth PDE-constrained optimization. *Inverse Problems and Imaging*, 7(4), 2013.
- [28] C.-A. Deledalle, S. Vaiter, J. Fadili, and G. Peyré. Stein unbiased gradient estimator of the risk (sugar) for multiple parameter selection. *SIAM Journal on Imaging Sciences*, 7(4):2448–2487, 2014.
- [29] H. Dirks. A flexible primal-dual toolbox. *ArXiv e-prints arXiv:1603.05835*, Mar. 2016.
- [30] D. C. Dobson and C. R. Vogel. Convergence of an iterative method for total variation denoising. *SIAM Journal on Numerical Analysis*, 34(5):1779–1791, 1997.
- [31] Y. Dong, M. Hintermüller, and M. Neri. An efficient primal-dual method for  $L^1$  TV image restoration. *SIAM Journal on Imaging Sciences*, 2(4):1168–1189, 2009.
- [32] Y. Dong, M. Hintermüller, and M. M. Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *Journal of Mathematical Imaging and Vision*, 40(1):82–104, 2011.
- [33] D. L. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224, 1995.
- [34] I. Ekeland and R. Témam. *Convex Analysis and Variational Problems*, volume 28 of *Classics in Applied Mathematics*. SIAM, Philadelphia, PA, english edition, 1999.
- [35] Y. C. Eldar. Generalized sure for exponential families: Applications to regularization. *Signal Processing, IEEE Transactions on*, 57(2):471–481, 2009.
- [36] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40, 1976.
- [37] R. Giryes, M. Elad, and Y. C. Eldar. The projected GSURE for automatic parameter tuning in iterative shrinkage methods. *Applied and Computational Harmonic Analysis*, 30(3):407–422, 2011.
- [38] E. Giusti. *Minimal Surfaces and Functions of Bounded Variation*, volume 80 of *Monographs in Mathematics*. Birkhäuser Verlag, Basel, 1984.
- [39] R. Glowinski. *Numerical Methods for Nonlinear Variational Problems*. Springer-Verlag, New York, 1984.
- [40] T. Goldstein and S. Osher. The split Bregman method for  $L_1$ -regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.
- [41] M. Hintermüller and K. Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM Journal on Applied Mathematics*, 64(4):1311–1333, 2004.
- [42] M. Hintermüller and A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1/L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.
- [43] M. Hintermüller, C. Rautenberg, T. Wu, and A. Langer. Optimal selection of the regularization function in a generalized total variation model. part II: Algorithm, its analysis and numerical tests. *Journal of Mathematical Imaging and Vision*, 59(3):515–533, 2017.

## 9. Box-Constraint Total Variation

- [44] M. Hintermüller and M. M. Rincon-Camacho. Expected absolute value estimators for a spatially adapted regularization parameter choice rule in  $L^1$ -TV-based image restoration. *Inverse Problems*, 26(8):085005, 30, 2010.
- [45] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.
- [46] K. Ito and K. Kunisch. An active set strategy based on the augmented Lagrangian formulation for image restoration. *ESAIM: Mathematical Modelling and Numerical Analysis*, 33(1):1–21, 1999.
- [47] K. Kunisch and T. Pock. A bilevel optimization approach for parameter learning in variational models. *SIAM Journal on Imaging Sciences*, 6(2):938–983, 2013.
- [48] A. Langer. Automated parameter selection for total variation minimization in image restoration. *Journal of Mathematical Imaging and Vision*, 57(2):239–268, 2017.
- [49] A. Langer. Automated parameter selection in the  $L^1$ - $L^2$ -TV model for removing Gaussian plus impulse noise. *Inverse Problems*, 33(7):074002, 2017.
- [50] A. Langer. Locally adaptive total variation for removing mixed Gaussian-impulse noise. *International Journal of Computer Mathematics*, pages 1–19, 2018.
- [51] Y. Lin, B. Wohlberg, and H. Guo. UPRE method for total variation parameter selection. *Signal Processing*, 90(8):2546–2551, 2010.
- [52] R. W. Liu, L. Shi, S. C. H. Yu, and D. Wang. Box-constrained second-order total generalized variation minimization with a combined  $L^{1,2}$  data-fidelity term for image reconstruction. *Journal of Electronic Imaging*, 24(3):033026–033026, 2015.
- [53] L. Ma, M. Ng, J. Yu, and T. Zeng. Efficient box-constrained TV-type- $l^1$  algorithms for restoring images with impulse noise. *Journal of Computational Mathematics*, 31:249–270, 2013.
- [54] C. L. Mallows. Some comments on  $C_P$ . *Technometrics*, 15(4):661–675, 1973.
- [55] B. Morini, M. Porcelli, and R. H. Chan. A reduced Newton method for constrained linear least-squares problems. *Journal of Computational and Applied Mathematics*, 233(9):2200–2212, 2010.
- [56] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1, Ser. A):127–152, 2005.
- [57] M. K. Ng, P. Weiss, and X. Yuan. Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods. *SIAM Journal on Scientific Computing*, 32(5):2710–2736, 2010.
- [58] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40(3):965–994, 2002.
- [59] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1-2):99–120, 2004.
- [60] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.
- [61] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.
- [62] M. Storath, C. Brandt, M. Hofmann, T. Knopp, J. Salamon, A. Weber, and A. Weinmann. Edge preserving and noise reducing reconstruction for magnetic particle imaging. *IEEE Transactions on Medical Imaging*, 36(1):74–85, 2017.
- [63] C. R. Vogel. *Computational Methods for Inverse Problems*, volume 23. SIAM, 2002.

## **II. Parameter Selection Methods for Total Variation Models**

- [64] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
- [65] P. Weiss, L. Blanc-Féraud, and G. Aubert. Efficient schemes for total variation minimization under constraints in image processing. *SIAM Journal on Scientific Computing*, 31(3):2047–2080, 2009.
- [66] B. M. Williams, K. Chen, and S. P. Harding. A new constrained total variational deblurring model and its fast algorithm. *Numerical Algorithms*, 69(2):415–441, 2015.

# 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model for Removing Gaussian plus Impulse Noise

**Bibliographic Note:** The content of this chapter is published in [AL9]: A. Langer. Automated parameter selection in the  $L^1$ - $L^2$ -TV model for removing Gaussian plus impulse noise. *Inverse Problems*, 33:41, 2017.

**Summary:** The minimization of a functional consisting of a combined  $L^1$ / $L^2$ -data-fidelity term and a total variation term, named  $L^1$ - $L^2$ -TV model, is considered to remove a mixture of Gaussian and impulse noise in images, which are possibly additionally deformed by some convolution operator. We investigate analytically the stability of this model with respect to its parameters and link it to a constrained minimization problem. Based on these investigations and a statistical characterization of the mixed Gaussian-impulse noise a fully automated parameter selection algorithm for the  $L^1$ - $L^2$ -TV model is presented. It is shown by numerical experiments that the proposed method finds parameters with which noise is removed considerably while features are preserved in images.

## 10.1. Introduction

Total variation as regularization in image restoration was first introduced in [84] and has received considerable attention in image processing. This is in particular due to its ability to preserve edges in images [24, 32]. In this context, one typically minimizes a functional that consists of a data-fidelity term, which enforces the consistency between the recovered and the measured image, and the total variation as a regularization term. The choice of the data term typically depends on the type of noise affecting the measured image. Usually images are corrupted by different types of noise, such as Gaussian noise, Poisson noise, and impulse noise. This contamination usually happens during image acquisition, which describes the process of capturing an image by a camera and converting it into a measurable entity [76], and image transmission. If no data is lost, i.e., the image is not affected by impulse noise, then mixed Poisson-Gaussian noise can be transformed into additive white Gaussian noise [50]. This might be the reason why most of the literature is solely dedicated to Gaussian denoising. Further Gaussian-Poisson noise removal strategies can be found in [11, 33, 54, 64, 65] and references therein. The task of removing Gaussian noise has been successfully performed by using a quadratic  $L^2$ -data-fidelity term in first order methods, see e.g. [23, 25, 26, 31, 37, 41, 42, 43, 46, 57, 77, 81, 89, 94, 14], as well as in second order methods, see e.g. [62]. In this approach, which we refer to as the  $L^2$ -TV model, the original image  $\hat{u}$  is recovered from the observed data  $g$  by solving

$$\min_{u \in BV(\Omega)} \alpha \|Tu - g\|_{L^2(\Omega)}^2 + |Du|(\Omega), \quad (10.1)$$

## II. Parameter Selection Methods for Total Variation Models

where  $\Omega \subset \mathbb{R}^2$  is an open bounded set with Lipschitz boundary,  $T$  is a bounded linear operator modeling the image-formation device (if the image is only corrupted by noise one sets  $T = I$ ), and  $\alpha > 0$  is a parameter. We recall, that for  $u \in L^1(\Omega)$

$$V(u, \Omega) := \sup \left\{ \int_{\Omega} u \operatorname{div} \phi dx : \phi \in [C_c^1(\Omega)]^2, \|\phi\|_{L^\infty(\Omega)} \leq 1 \right\}$$

is the variation of  $u$  in  $\Omega$ . Here,  $L^q(\Omega)$ , with  $q \in [1, \infty]$ , denotes the usual Lebesgue space [2] and  $C_c^l(\Omega)$ ,  $l \in \mathbb{N}$ , is the space of  $l$ -times continuously differentiable functions with compact support in  $\Omega$ . In the event that  $V(u, \Omega) < \infty$  we denote  $|Du|(\Omega) = V(u, \Omega)$  and call it the total variation of  $u$  in  $\Omega$ ; see [5, 55] for more details. If  $u \in W^{1,1}(\Omega)$ , then  $|Du|(\Omega) = \int_{\Omega} |\nabla u| dx$ . Further,  $BV(\Omega)$  denotes the space of functions with bounded variation, i.e.,  $u \in BV(\Omega)$  if and only if  $V(u, \Omega) < \infty$ . The space  $BV(\Omega)$  endowed with the norm  $\|u\|_{BV} = \|u\|_{L^1(\Omega)} + |Du|(\Omega)$  is a Banach space [55].

Other efficient Gaussian denoising approaches can be found, for example, in [39, 40, 69] and references therein.

The  $L^2$ -TV model usually does not yield a satisfactory restoration in the presence of impulse noise. This type of noise is usually constituted due to malfunctioning pixels in camera sensors, faulty memory locations in hardware, or transmission over noisy digital links. There are two commonly used models of impulse noise considered in the literature. The first one, called *salt-and-pepper noise*, where the noisy image  $g$  is given by

$$g(x) = \begin{cases} 0 & \text{with probability } s_1 \in [0, 1], \\ 1 & \text{with probability } s_2 \in [0, 1], \\ T\hat{u}(x) & \text{with probability } 1 - s_1 - s_2, \end{cases}$$

with  $1 - s_1 - s_2 > 0$  [29]. Here and in the rest of the paper we assume that  $T\hat{u}$  is in the dynamic range  $[0, 1]$ , i.e.,  $0 \leq T\hat{u} \leq 1$ . The second model is called *random-valued impulse noise*, where  $g$  is described as

$$g(x) = \begin{cases} c & \text{with probability } s \in [0, 1], \\ T\hat{u}(x) & \text{with probability } 1 - s, \end{cases}$$

with  $c$  being a uniformly distributed random variable in the image intensity range  $[0, 1]$ . For impulse noise contaminated images a more successful approach uses instead of a quadratic  $L^2$ -data-fidelity term a nonsmooth  $L^1$ -data-fidelity term [4, 79, 80]. That is, instead of (10.1) one optimizes the following minimization problem

$$\min_{u \in BV(\Omega)} \alpha \|Tu - g\|_{L^1(\Omega)} + |Du|(\Omega), \quad (10.2)$$

which we call the  $L^1$ -TV model.

Instead of assuming that an image is only contaminated by one type of noise, in this paper we consider a mixture of Gaussian and impulse noise. Recently in [60] an  $L^1$ - $L^2$ -data-fidelity term has been introduced and shown to be suited to the task of removing mixed Gaussian-impulse noise. In this approach, which we call  $L^1$ - $L^2$ -TV model, an image is restored by solving

$$\min_{u \in BV(\Omega)} \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + |Du|(\Omega), \quad (10.3)$$

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

where  $T_i : L^2(\Omega) \rightarrow L^2(\Omega)$  is a bounded linear operator,  $g_i \in L^2(\Omega)$  is a given datum, and  $\alpha_i \geq 0$  for  $i = 1, 2$  with  $\alpha_1 + \alpha_2 > 0$ . For the case of removing a mixture of Gaussian and impulse noise from an image  $g$  one typically sets  $T_1 = T_2$  and  $g_1 = g_2 = g$  in (10.3). In this setting it is easy to see that the  $L^1$ - $L^2$ -TV model (10.3) is a generalization of (10.1) and (10.2). In particular, if we set  $\alpha_2 = 0$  in (10.3) then we obtain the  $L^1$ -TV model while for  $\alpha_1 = 0$  we obtain the  $L^2$ -TV model. Modifications of the  $L^1$ - $L^2$ -TV model have been presented in [58], where the total variation is replaced by  $\|Wu\|_1$  with  $W$  being a wavelet tight frame transform, and in [73], where the second order total generalized variation [18] has been used as regularization term and box-constraints, which assure that the reconstruction lies in the respective dynamic range, are incorporated. We also note, that for impulse noise-dominated contamination of image data the implementation of an impulse noise detector, such as the one in [28] and the references therein, enhance the model.

Other approaches for removing mixed Gaussian-impulse noise studied in the literature usually start by estimating or detecting outliers (impulse noise) in the image and then adapt or use a Gaussian noise removal; see for example [20, 53, 63, 71, 91, 93]. In general, algorithms for Gaussian plus impulse noise removal may be classified in the following way: filter approaches [52, 82, 93], regularization based approaches [20, 47, 53, 63, 71, 85, 91, 92], Bayesian-based approaches [74], and patch-based approaches [45, 70, 72].

The  $L^1$ - $L^2$ -TV model and its aforementioned modifications clearly fall into the class of regularization based approaches and (the restoration quality of) its solution highly depends on the proper choice of  $\alpha_i$ ,  $i = 1, 2$ . In particular, small  $\alpha_1$  and  $\alpha_2$ , which lead to an over-smoothed reconstruction, not only remove noise but also eliminate details in the image. On the contrary, large  $\alpha_1$  and  $\alpha_2$  lead to solutions that fit the given data properly but retain noise. Note, that  $\alpha_1$  and  $\alpha_2$  weight the importance of the  $L^1$ -term and  $L^2$ -term. In particular, we expect  $\alpha_1$  to be large if the noise in the image is impulse noise dominated, while for Gaussian noise dominated images  $\alpha_2$  should be sufficiently large. Hence a good reconstruction can be achieved by choosing  $\alpha_1$  and  $\alpha_2$  such that a good compromise of the aforementioned effects are made. In [73] it is suggested to select the parameters according to the variance  $\sigma^2$  of the Gaussian noise and the energy of the impulse noise, i.e.,

$$\alpha_1 = \frac{E_I}{E_I + \sigma^2} \quad \text{and} \quad \alpha_2 = \frac{\sigma^2}{E_I + \sigma^2}, \quad (10.4)$$

where  $E_I = \frac{s_1+s_2}{2}$  for salt-and-pepper noise and  $E_I = \frac{s}{3}$  for random-valued impulse noise. It is demonstrated in [73] that by setting the parameters according to (10.4) suitable restorations are obtained. Note, that this parameter selection depends on the noise-levels of the different contained noises without using the statistical behavior of mixed Gaussian-impulse noise.

For more general problems including (10.3) with  $T_1 = T_2 = I$  in [44] based on a training set of pairs  $(g_k, \hat{u}_k)$ ,  $k = 1, 2, \dots, N \in \mathbb{N}$ , where  $g_k$  is the noisy image and  $\hat{u}_k$  represents the original image, a bilevel optimization approach is presented, which computes suitable parameters of the corresponding image model. Similar approaches are also discussed in [21, 67] and references therein. However, since in our setting we do not have a training set given, these approaches are not applicable here.

In this paper we are investigating the statistical characterization of mixed Gaussian-impulse noise, where we assume that  $\sigma, s_1, s_2$  and  $s$  are at hand, and use it to formulate a fully automated parameter adjustment strategy based on the discrepancy principle to compute suitable  $\alpha_1$  and  $\alpha_2$  for (10.3). In order to construct such a method, we link the  $L^1$ - $L^2$ -TV model with a constrained minimization problem consisting of two constraints, one related

## II. Parameter Selection Methods for Total Variation Models

to the  $L^1$ -term and the other related to the  $L^2$ -term. In particular, there exist  $\alpha_1$  and  $\alpha_2$  such that a solution of the constrained minimization problem is also a minimizer of the  $L^1$ - $L^2$ -TV model; see Theorem 10.5 below. Based on the discrepancy principle we suggest an iterative adjustment scheme, which utilizes an interplay between the penalized problem (10.3) and the associated constrained problem in order to either increase or decrease the parameter  $\alpha_i$ ,  $i = 1, 2$ , in each iteration. Since this update rule generates monotonic sequences of parameters, we are able to show that the proposed method indeed converges. Moreover, in each iteration the  $L^1$ - $L^2$ -TV model has to be solved with the current parameters. An algorithm for solving such a minimization problem is presented in [60] without any theoretical justification of its convergence. Here we use the same algorithm and provide a convergence proof. In our numerical experiments we demonstrate that the proposed automated parameter selection method indeed finds parameters  $\alpha_1$  and  $\alpha_2$  such that the corresponding restoration is better with respect to some restoration quality measure than the one obtained with (10.4).

Instead of looking for suitable parameters  $\alpha_1$  and  $\alpha_2$  one may solve the associated constrained optimization problem directly. In this vein we utilize the alternating direction method to compute a numerical minimizer of the constrained problem and compare it with the automated parameter selection algorithm.

The rest of the paper is organized as follows: In Section 10.2 a statistical characterization of mixed Gaussian-impulse noise is given. The link between the  $L^1$ - $L^2$ -TV model and a constrained minimization problem is investigated in Section 10.3. In this context, a collection of interesting properties of the  $L^1$ - $L^2$ -TV model is given. For example we prove a stability result of its minimizers with respect to the parameters  $\alpha_1$  and  $\alpha_2$ . Based on the constrained minimization problem together with the statistical characterization of the noise in Section 10.4 our proposed parameter selection algorithm is presented. This algorithm requires in each iteration the solution of the  $L^1$ - $L^2$ -TV model for which a solution algorithm is stated in Section 10.5 together with its convergence properties. In Section 10.6 we show numerical experiments which demonstrate that the proposed algorithm indeed finds parameters  $\alpha_1$  and  $\alpha_2$  that provide a good compromise of the effects described above. Finally in Section 10.7 conclusions are drawn.

### 10.2. Statistical Characterization of the Noise

Impulse noise (e.g. salt-and-pepper noise and random-valued impulse noise) is in general nonadditive and an observation  $g$  might be modeled as  $g = \mathcal{N}(T\hat{u})$ , where  $\mathcal{N}$  represents impulse noise. However, the corruption produced by  $\mathcal{N}$  is  $g - T\hat{u} =: \rho_{\hat{u}}$  and hence we may view it in an additive fashion, i.e.,  $g = T\hat{u} + \rho_{\hat{u}}$ . In this vein at a point  $x \in \Omega$  the Gaussian and impulse noise contaminated image might be written as  $g(x) = T\hat{u}(x) + \eta_{\hat{u}}(x) + \rho_{\hat{u}}(x)$ , which is a stochastic observation, where the random values  $\eta_{\hat{u}}(x)$  and  $\rho_{\hat{u}}(x)$  depend on the underlying noise. In particular, the random element  $\eta_{\hat{u}}$  represents Gaussian noise with zero mean and variance  $\sigma^2$ , while  $\rho_{\hat{u}}$  represents salt-and-pepper noise or random-valued impulse noise. The processes of contaminating an image by Gaussian noise and impulse noise are here assumed to be independent from each other, which seems natural, since usually Gaussian and impulse noise are constituted from different physical processes. For example, due to image registration Gaussian noise is added and later digital transmission adds impulse noise. Moreover, for any two points  $x, y \in \Omega$  we assume that  $\eta_{\hat{u}}(x)$  and  $\eta_{\hat{u}}(y)$  as well as  $\rho_{\hat{u}}(x)$  and  $\rho_{\hat{u}}(y)$  are independent, cf. [61]. By analogous considerations as in [61] we obtain the following characterizations.

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

In the following  $X$  and  $Y$  denote random variables and  $\mathbb{P}$  denotes the probability of an event, e.g.,  $\mathbb{P}(X = 0)$  denotes the probability of  $X$  being 0.

**Gaussian Noise** For  $\eta_{\hat{u}}$  being normally distributed with zero mean and standard deviation  $\sigma$  the mean ( $\mathbb{E}$ ), variance ( $\text{Var}$ ), and expected absolute value ( $\text{EAV}$ ) are

$$\mathbb{E}(\eta_{\hat{u}}) = 0, \quad \text{Var}(\eta_{\hat{u}}) = \sigma^2, \quad \text{and} \quad \text{EAV}(\eta_{\hat{u}}) = \sqrt{\frac{2}{\pi}}\sigma.$$

**Salt-and-Pepper Noise** Let  $X$  be a random variable distributed according to the probability density function of salt-and-pepper noise given as

$$f_{sp}(X \mid \hat{u}) = \begin{cases} 1 - s_1 - s_2 & \text{if } X = 0, \\ s_1 & \text{if } X = -T\hat{u}, \\ s_2 & \text{if } X = 1 - T\hat{u}. \end{cases}$$

If  $\rho_{\hat{u}}$  represents salt-and-pepper noise, then the expected value of  $\rho_{\hat{u}}$  calculates as

$$\mathbb{E}(\rho_{\hat{u}} \mid \hat{u}) = 0 \cdot \mathbb{P}(\rho_{\hat{u}} = 0) - (T\hat{u}) \cdot \mathbb{P}(\rho_{\hat{u}} = -T\hat{u}) + (1 - T\hat{u}) \cdot \mathbb{P}(\rho_{\hat{u}} = 1 - T\hat{u}) = s_2(1 - T\hat{u}) - s_1T\hat{u}.$$

For the variance and the expected absolute value we obtain

$$\begin{aligned} \text{Var}(\rho_{\hat{u}} \mid \hat{u}) &= \mathbb{E}(\rho_{\hat{u}}^2) - \mathbb{E}(\rho_{\hat{u}})^2 = s_2(1 - T\hat{u})^2 + s_1(T\hat{u})^2 - (s_2(1 - T\hat{u}) - s_1T\hat{u})^2, \\ \text{EAV}(\rho_{\hat{u}} \mid \hat{u}) &:= \mathbb{E}(|\rho_{\hat{u}}| \mid \hat{u}) = s_2(1 - T\hat{u}) + s_1T\hat{u}. \end{aligned}$$

Assuming that the range of  $T\hat{u}$  belongs to the interval  $[0, 1]$ , we find

$$\mathbb{E}(\rho_{\hat{u}} \mid \hat{u}) \in [-s_1, s_2], \quad \text{Var}(\rho_{\hat{u}} \mid \hat{u}) \in \left[ \frac{s_2s_1^2 + s_2^2s_1}{(s_1 + s_2)^2}, \max\{s_1 - s_1^2, s_2 - s_2^2\} \right],$$

and

$$\text{EAV}(\rho_{\hat{u}} \mid \hat{u}) \in [\min\{s_1, s_2\}, \max\{s_1, s_2\}].$$

**Random-Valued Impulse Noise** In case of random-valued impulse noise the probability density function is described as

$$f_{rv}(X \mid \hat{u}) = \begin{cases} 1 - s & \text{if } X = 0, \\ s & \text{if } X = Y - T\hat{u}, \end{cases}$$

where  $Y$  is a uniformly distributed random variable in the interval  $[0, 1]$ . Then we get for the random variable  $\rho_{\hat{u}}$

$$\begin{aligned} \mathbb{E}(\rho_{\hat{u}} \mid \hat{u}) &= \int_0^1 s(Y - T\hat{u})dY = s\left(\frac{1}{2} - T\hat{u}\right), \\ \text{Var}(\rho_{\hat{u}} \mid \hat{u}) &= \int_0^1 s(Y - T\hat{u})^2dY - s^2\left(\frac{1}{2} - T\hat{u}\right)^2 = s\left(\frac{1}{3} - T\hat{u} + (T\hat{u})^2\right) - s^2\left(\frac{1}{2} - T\hat{u}\right)^2, \\ \text{EAV}(\rho_{\hat{u}} \mid \hat{u}) &= \int_0^1 s|Y - T\hat{u}|dY = \int_{T\hat{u}}^1 s(Y - T\hat{u})dY + \int_0^{T\hat{u}} s(T\hat{u} - Y)dY \\ &= s\left(\frac{1}{2} - T\hat{u} + (T\hat{u})^2\right). \end{aligned}$$

## II. Parameter Selection Methods for Total Variation Models

Since  $T\hat{u} \in [0, 1]$ , we have

$$\mathbb{E}(\rho_{\hat{u}} | \hat{u}) \in \left[-\frac{s}{2}, \frac{s}{2}\right], \text{ EAV}(\rho_{\hat{u}} | \hat{u}) \in \left[\frac{s}{4}, \frac{s}{2}\right], \text{ and } \text{Var}(\rho_{\hat{u}} | \hat{u}) \in \left[\frac{s}{12}, \frac{s}{3} - \frac{s^2}{4}\right].$$

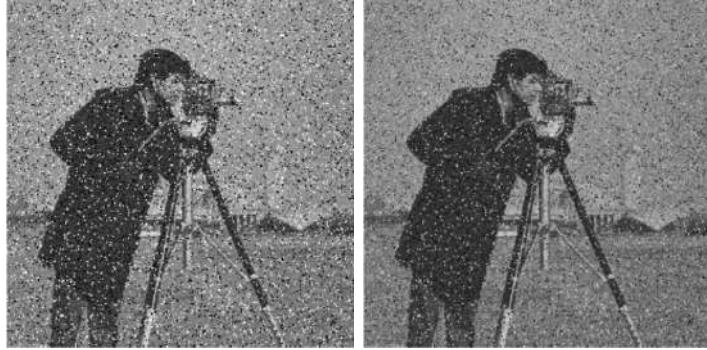
**Mixed Noise** Since  $\eta_{\hat{u}}$  and  $\rho_{\hat{u}}$  are independent random variables, we obtain for a combination of Gaussian and impulse noise that the variance is given by

$$\nu_2 = \nu_2(\hat{u}) := \text{Var}(\eta_{\hat{u}} + \rho_{\hat{u}} | \hat{u}) = \text{Var}(\eta_{\hat{u}}) + \text{Var}(\rho_{\hat{u}} | \hat{u}) \quad (10.5)$$

while the expected absolute value can be estimated from below and above by

$$|\mathbb{E}(\eta_{\hat{u}} + \rho_{\hat{u}} | \hat{u})| \leq \nu_1 = \nu_1(\hat{u}) := \text{EAV}(\eta_{\hat{u}} + \rho_{\hat{u}} | \hat{u}) \leq \text{EAV}(\eta_{\hat{u}}) + \text{EAV}(\rho_{\hat{u}} | \hat{u}). \quad (10.6)$$

Note, that there is a difference whether Gaussian noise is added before or after the impulse noise. This is due to the definition of impulse noise. For example, salt-and-pepper noise sets with a certain probability a value of a pixel to the minimal or maximal value of the image intensity range. This intensity range may be extended due to Gaussian noise. Hence, the observation may be different depending which noise is added first, see Figure 10.1.



**Figure 10.1:** “cameraman” image, see Figure 10.5(a), corrupted by (left) first Gaussian noise and then salt-and-pepper noise (right) first salt-and-pepper noise and then Gaussian noise.

If an image is first contaminated by impulse noise then, since  $T\hat{u} \in [0, 1]$  and hence  $T\hat{u} + \rho_{\hat{u}} \in [0, 1]$ , we obtain by the above estimates the following bounds:

$$\begin{aligned} \text{Gaussian + salt-and-pepper: } & \nu_1 \in \left[0, \sqrt{\frac{2}{\pi}}\sigma + \max\{s_1, s_2\}\right] \\ & \nu_2 \in \sigma^2 + \left[\frac{s_1 s_2^2 + s_1^2 s_2}{(s_1 + s_2)^2}, \max\{s_1 - s_1^2, s_2 - s_2^2\}\right]. \\ \text{Gaussian + random-valued: } & \nu_1 \in \left[0, \sqrt{\frac{2}{\pi}}\sigma + \frac{s}{2}\right] \\ & \nu_2 \in \sigma^2 + \left[\frac{s}{12}, \frac{s}{3} - \frac{s^2}{4}\right]. \end{aligned} \quad (10.7)$$

If Gaussian noise is added first, then by rescaling the image such that  $T\hat{u} + \eta_{\hat{u}} \in [0, 1]$  the same bounds as in (10.7) are obtained. We remark, that the above calculations of the variance and expected absolute value can be adjusted to the situation  $T\hat{u} \in [c_{\min}, c_{\max}]$ ,  $c_{\min} < c_{\max}$ , which would make a rescaling of the image unnecessary to obtain bounds similar to (10.7).

### 10.3. Constrained versus Unconstrained Problem

We define the functional in (10.3) as

$$\mathcal{J}_{\alpha_1, \alpha_2}(u) := \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + |Du|(\Omega)$$

and link the optimization problem (10.3) to the constrained minimization problem

$$\min_{u \in BV(\Omega)} |Du|(\Omega) \quad \text{subject to (s.t.)} \quad \|T_1 u - g_1\|_{L^1(\Omega)} \leq \nu_1 |\Omega| \quad \text{and} \quad \|T_2 u - g_2\|_{L^2(\Omega)}^2 \leq \nu_2 |\Omega|, \quad (10.8)$$

where  $\nu_1, \nu_2 \geq 0$  denote the expected absolute value and the variance of the underlying noise, respectively. Here, we assume that  $\nu_1$  and  $\nu_2$  are fixed constants in the intervals as specified in (10.7). However, in our numerical experiments we report on results where  $\nu_1$  and  $\nu_2$  are chosen empirically based on some approximation of the true image. If  $g_1 = g_2$ , then we easily see from the previous section that  $\nu_1$  and  $\nu_2$  are correlated by the statistical values  $\sigma, s_1, s_2$ , and  $s$  of the noise. For example, if  $\nu_1 = 0$ , then also  $\nu_2 = 0$  and hence no noise is present. Note, that in the general case where  $g_1 \neq g_2$  is allowed, such a correlation might not be valid.

It is worth to mention, that an analysis of the relation between convex constrained minimization problems with solely one constraint and their associated penalized problem can be found in [6, 34] and references therein.

#### 10.3.1. Existence of Minimizers

For showing existence of a solution of (10.8) we start by adapting a result of [1].

**Lemma 10.1.** *Assume there exist  $i \in \{1, 2\}$  such that  $T_i$  does not annihilate constant functions, i.e.,  $T_i \chi_\Omega \neq 0$ , where  $\chi_\Omega(x) = 1$  if  $x \in \Omega$ . Then  $\|u\|_{BV} \rightarrow \infty$  implies  $\mathcal{J}_{1,1}(u) \rightarrow \infty$ .*

*Proof.* Since  $\mathcal{J}_{1,1}(u) \geq |Du|(\Omega) + \|T_i u - g_i\|_{L^i(\Omega)}^i$  for  $i = 1, 2$ , the assertion immediately follows from [1, Lemma 4.1] and [61, Proposition 1].  $\square$

Next, we define the feasible set

$$U := \{u \in BV(\Omega) : \|T_1 u - g_1\|_{L^1(\Omega)} \leq \nu_1 |\Omega| \text{ and } \|T_2 u - g_2\|_{L^2(\Omega)}^2 \leq \nu_2 |\Omega|\}.$$

Moreover, the convex problem (10.8) is *superconsistent* if there is a feasible point  $u$  of the problem such that  $\|T_i u - g_i\|_{L^i(\Omega)}^i < \nu_i |\Omega|$  for  $i = 1, 2$  [83]. Note, that if  $g_1 = g_2$  and  $T_1 = T_2$ , which is the most relevant case for removing a mixture of Gaussian-impulse noise, we have that  $U \neq \emptyset$  and (10.8) is superconsistent, cf. for example [12, 26]. On the contrary, if  $g_1 \neq g_2$ , then the feasible set might be even empty. However, for example an assumption like  $\nu_1 |\Omega| > \|g_1\|_{L^1(\Omega)}$  and  $\nu_2 |\Omega| > \|g_2\|_{L^2(\Omega)}^2$  would guarantee the nonemptiness of  $U$  and the superconsistency of (10.8). For the sake of generality, in the sequel we will just assume that the set  $U$  is not empty or even that the problem (10.8) is superconsistent.

Now we are able to argue the existence of a minimizer of (10.8).

**Theorem 10.2.** *Assume there exist  $i \in \{1, 2\}$  such that  $T_i$  does not annihilate constant functions and  $U \neq \emptyset$ . Then the problem in (10.8) has a solution  $u \in BV(\Omega)$ .*

## II. Parameter Selection Methods for Total Variation Models

*Proof.* Choose an infimal sequence  $(u_n)_n \subset U$  of (10.8). Lemma 10.1 yields that  $(u_n)_n$  is bounded in  $BV(\Omega)$ . Then there exists a subsequence  $(u_{n_k})_k \subset U$  which converges weakly in  $L^2(\Omega)$  to some  $u^* \in L^2(\Omega)$ . The lower semicontinuity of the total variation  $|D \cdot|(\Omega)$  with respect to the  $L^2(\Omega)$  topology [1, Theorem 2.3] implies  $u^* \in BV(\Omega)$ . The sequence  $(Du_{n_k})_k$  converges weakly as a measure to  $Du^*$  [1, Lemma 2.5]. Since  $T_1$  and  $T_2$  are continuous linear operators,  $(T_i u_{n_k})_k$  converges weakly to  $T_i u^*$  in  $L^2(\Omega)$ . By the lower semicontinuity we have

$$\begin{aligned}\|T_1 u^* - g_1\|_{L^1(\Omega)} &\leq \liminf_{k \rightarrow \infty} \|T_1 u_{n_k} - g_1\|_{L^1(\Omega)} \leq \nu_1 |\Omega| \\ \|T_2 u^* - g_2\|_{L^2(\Omega)}^2 &\leq \liminf_{k \rightarrow \infty} \|T_2 u_{n_k} - g_2\|_{L^2(\Omega)}^2 \leq \nu_2 |\Omega|\end{aligned}$$

and hence  $u^* \in BV(\Omega)$  is a solution of (10.8).  $\square$

The assumption that at least either  $T_1$  or  $T_2$  does not annihilating constant functions also ensures that  $\mathcal{J}_{\alpha_1, \alpha_2}$  has a minimizer if  $\alpha_1, \alpha_2 > 0$ . In particular, we have the following result.

**Theorem 10.3.** *Assume there exist  $i \in \{1, 2\}$  such that  $T_i$  does not annihilate constant functions and  $\alpha_i > 0$ . Then the problem in (10.3) has a solution  $u \in BV(\Omega)$ . If  $\alpha_2 > 0$  and  $T_2$  is injective, then the minimizer  $u$  is unique.*

*Proof. Existence:* The existence of a solution of (10.3) follows from the same arguments as the ones from the proof of Theorem 10.2 by noting that the lower semicontinuity yields

$$\mathcal{J}_{\alpha_1, \alpha_2}(u^*) \leq \liminf_{k \rightarrow \infty} \mathcal{J}_{\alpha_1, \alpha_2}(u_{n_k}),$$

where  $u_{n_k}$  is a subsequence of a minimizing sequence for  $\mathcal{J}_{\alpha_1, \alpha_2}$  and  $u^*$  is its limit. Consequently  $u^* \in BV(\Omega)$  is a minimizer of  $\mathcal{J}_{\alpha_1, \alpha_2}$ .

*Uniqueness:* If  $\alpha_2 > 0$ , then similar as in the proof of [87, Proposition 3.1], let  $u, v \in BV(\Omega)$  be two minimizers of  $\mathcal{J}_{\alpha_1, \alpha_2}$  and  $T_2 u \neq T_2 v$ . Then by the strict convexity of the  $L^2$ -term we get

$$\mathcal{J}_{\alpha_1, \alpha_2}\left(\frac{u+v}{2}\right) < \frac{1}{2}\mathcal{J}_{\alpha_1, \alpha_2}(u) + \frac{1}{2}\mathcal{J}_{\alpha_1, \alpha_2}(v) = \min_{w \in BV(\Omega)} \mathcal{J}_{\alpha_1, \alpha_2}(w).$$

Since  $u$  and  $v$  are minimizers, this inequality cannot be true, and hence  $T_2 u = T_2 v$ . If  $T_2$  is injective, then we have  $u = v$ .  $\square$

Note, that if  $\alpha_2 > 0$  and  $T_2$  is injective, then  $\alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2$  and hence  $\mathcal{J}_{\alpha_1, \alpha_2}$  is strictly convex, which renders its minimizer unique. The uniqueness of minimizers can be also obtained by the following stability result, cf. [9, Theorem 10.6] for the  $L^2$ -TV model with  $T_2 = I$ .

**Proposition 10.4.** *For  $g_1, g_2, f_1, f_2 \in L^2(\Omega)$  let the functions  $u_g, u_f \in BV(\Omega)$  be minimizers of*

$$\min_{u \in BV(\Omega)} \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + |Du|(\Omega)$$

and

$$\min_{u \in BV(\Omega)} \alpha_1 \|T_1 u - f_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - f_2\|_{L^2(\Omega)}^2 + |Du|(\Omega),$$

respectively. Then for  $\alpha_2 > 0$  and  $\alpha_1 \geq 0$  we have that

$$\|T_2(u_f - u_g)\|_{L^2(\Omega)} \leq \frac{1}{2} \|f_2 - g_2\|_{L^2(\Omega)} + \frac{1}{2\alpha_2} \sqrt{\alpha_2^2 \|f_2 - g_2\|_{L^2(\Omega)}^2 + 4\alpha_1\alpha_2 \|f_1 - g_1\|_{L^1(\Omega)}}.$$

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

*Proof.* Define the convex functionals  $G_g(u) := \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2$ ,  $G_f(u) := \alpha_2 \|T_2 u - f_2\|_{L^2(\Omega)}^2$ ,  $F_g(u) := \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + |Du|(\Omega)$ ,  $F_f(u) := \alpha_1 \|T_1 u - f_1\|_{L^1(\Omega)} + |Du|(\Omega)$  and set  $\mathcal{J}_g(u) := G_g(u) + F_g(u)$  and  $\mathcal{J}_f(u) := G_f(u) + F_f(u)$ . We extend  $F_g$  and  $F_f$  to  $L^2(\Omega)$  with the value  $+\infty$ . Moreover, we note that  $G_g$  and  $G_f$  are Fréchet differentiable.

For  $x \in \partial F_g(u)$  and  $y \in \partial F_f(v)$  we have by the definition of subdifferential, see for example [49], that

$$\begin{aligned} F_g(w) &\geq F_g(u) + \langle x, w - u \rangle \quad \text{for all } w \in L^2(\Omega), \\ F_f(\tilde{w}) &\geq F_f(v) + \langle y, \tilde{w} - v \rangle \quad \text{for all } \tilde{w} \in L^2(\Omega). \end{aligned}$$

Summing up these inequalities for  $w = v$  and  $\tilde{w} = u$  yields

$$\begin{aligned} \langle x - y, v - u \rangle &\leq \alpha_1 (\|T_1 v - g_1\|_{L^1(\Omega)} - \|T_1 v - f_1\|_{L^1(\Omega)} + \|T_1 u - f_1\|_{L^1(\Omega)} - \|T_1 u - g_1\|_{L^1(\Omega)}) \\ &\leq \alpha_1 (\|f_1 - g_1\|_{L^1(\Omega)} + \|f_1 - g_1\|_{L^1(\Omega)}). \end{aligned}$$

From this together with the optimality of  $u_g$  and  $u_f$ , i.e.,  $-\partial G_g(u_g) \in \partial F_g(u_g)$  and  $-\partial G_f(u_f) \in \partial F_f(u_f)$ , we obtain

$$2\alpha_2 \langle T_2(u_f - u_g) + g_2 - f_2, T_2(u_f - u_g) \rangle \leq 2\alpha_1 \|f_1 - g_1\|_{L^1(\Omega)}$$

which is equivalent to

$$2\alpha_2 \|T_2(u_f - u_g)\|_{L^2(\Omega)}^2 + 2\alpha_2 \langle g_2 - f_2, T_2(u_f - u_g) \rangle \leq 2\alpha_1 \|f_1 - g_1\|_{L^1(\Omega)}.$$

Using Hölder's inequality implies then

$$\alpha_2 \|T_2(u_f - u_g)\|_{L^2(\Omega)}^2 - \alpha_2 \|g_2 - f_2\|_{L^2(\Omega)} \|T_2(u_f - u_g)\|_{L^2(\Omega)} - \alpha_1 \|f_1 - g_1\|_{L^1(\Omega)} \leq 0.$$

This is a quadratic inequality in  $\|T_2(u_f - u_g)\|_{L^2(\Omega)}$  and hence calculating the roots yields

$$\|T_2(u_f - u_g)\|_{L^2(\Omega)} \leq \frac{1}{2\alpha_2} \left( \alpha_2 \|f_2 - g_2\|_{L^2(\Omega)} + \sqrt{\alpha_2^2 \|f_2 - g_2\|_{L^2(\Omega)}^2 + 4\alpha_1\alpha_2 \|f_1 - g_1\|_{L^1(\Omega)}} \right),$$

where we noted that  $\sqrt{\alpha_2^2 \|f_2 - g_2\|_{L^2(\Omega)}^2 + 4\alpha_1\alpha_2 \|f_1 - g_1\|_{L^1(\Omega)}} \geq \alpha_2 \|f_2 - g_2\|_{L^2(\Omega)}$ .  $\square$

Motivated by results in [12] we link the constrained minimization problem (10.8) to the unconstrained minimization problem (10.3).

**Theorem 10.5.** *Assume that  $T_i$  does not annihilate constant functions for  $i = 1, 2$  and (10.8) is superconsistent. Then there exists an  $\alpha = (\alpha_1, \alpha_2) \geq 0$  such that a solution of (10.3) satisfies the constraints in (10.8). Moreover, if  $\alpha_i > 0$  then  $\|T_i u - g_i\|_{L^i(\Omega)}^i = \nu_i |\Omega|$  for this value of  $i$ . In particular, there exist  $i \in \{1, 2\}$  such that  $\alpha_i > 0$ , if at least one of the following conditions holds:*

$$(C1) \inf_{c \in \mathbb{R}} \|g_1 - c\|_{L^1(\Omega)} > \nu_1 |\Omega| \quad \text{and} \quad T_1 \cdot 1 = 1$$

$$(C2) \inf_{c \in \mathbb{R}} \|g_2 - c\|_{L^2(\Omega)}^2 > \nu_2 |\Omega| \quad \text{and} \quad T_2 \cdot 1 = 1$$

## II. Parameter Selection Methods for Total Variation Models

*Proof.* We define the Lagrange function

$$L(u, \alpha) := |Du|(\Omega) + \alpha_1(\|T_1 u - g_1\|_{L^1(\Omega)} - \nu_1 |\Omega|) + \alpha_2(\|T_2 u - g_2\|_{L^2(\Omega)}^2 - \nu_2 |\Omega|).$$

Let  $u^* \in BV(\Omega)$  be a solution of (10.8), then, since the convex problem (10.8) is superconsistent, the Karush-Kuhn-Tucker Theorem [83, p. 182] yields that there exists an  $\alpha^* = (\alpha_1^*, \alpha_2^*) \geq 0$  such that

$$L(u^*, \alpha) \leq L(u^*, \alpha^*) \leq L(u, \alpha^*) \quad (10.9)$$

for all  $u \in BV(\Omega)$  and all  $\alpha = (\alpha_1, \alpha_2) \geq 0$ , and

$$\alpha_i^*(\|T_i u^* - g_i\|_{L^i(\Omega)}^i - \nu_i |\Omega|) = 0$$

for  $i = 1, 2$ . That is,  $\|T_i u - g_i\|_{L^i(\Omega)}^i = \nu_i |\Omega|$  if  $\alpha_i^* > 0$ . By the second inequality in (10.9) we see that  $u^*$  is also a minimizer of (10.3).

Let us finally show that not both  $\alpha_i = 0$ , if condition (C1) and/or (C2) holds. If  $\alpha_1 = \alpha_2 = 0$ , then for the associated solution  $\tilde{u}$  of (10.8) we would have that  $|D\tilde{u}|(\Omega) \leq |Du|(\Omega)$  for all  $u \in BV(\Omega)$  and hence  $\tilde{u} = c \in \mathbb{R}$  is constant. Assume condition (C1) holds, then  $T_1 \cdot 1 = 1$  and we obtain

$$\|g_1 - c\|_{L^1(\Omega)} = \|g_1 - T_1 \tilde{u}\|_{L^1(\Omega)} \leq \nu_1 |\Omega|$$

which is a contradiction to (C1). By the same arguments one shows the statement for (C2).  $\square$

### 10.3.2. Stability of the $L^1$ - $L^2$ -TV Model with Respect to Its Parameters

We define the minimum values of the energy  $\mathcal{J}_{\alpha_1, \alpha_2}$  by

$$\mathcal{E}(\alpha_1, \alpha_2) := \min_{u \in BV(\Omega)} \mathcal{J}_{\alpha_1, \alpha_2}(u).$$

Following [30] we obtain the following result.

**Proposition 10.6.** *For any given  $g_i \in L^2(\Omega)$  for  $i = 1, 2$  the function  $\mathcal{E}$  has the following properties:*

1.  $\mathcal{E}(0, 0) = 0$ .
2.  $0 \leq \mathcal{E}(\alpha_1, \alpha_2) \leq \alpha_1 \|g_1\|_{L^1(\Omega)} + \alpha_2 \|g_2\|_{L^2(\Omega)}^2$  for all  $\alpha_1, \alpha_2 \geq 0$ .

*Proof.* Since  $\mathcal{E}(0, 0) = \min_{u \in BV(\Omega)} \mathcal{J}_{0,0}(u) = \min_{u \in BV(\Omega)} |Du|(\Omega) = 0$ , the first statement follows. Further, we have  $0 \leq \mathcal{E}(\alpha_1, \alpha_2) \leq \mathcal{J}_{\alpha_1, \alpha_2}(0)$ , which shows the second statement.  $\square$

Similar as for the  $L^1$ -TV model and the  $L^2$ -TV model, see [30, 26], we have a monotonicity property of the data-fidelity terms with respect to the parameters  $\alpha_1$  and  $\alpha_2$ .

**Proposition 10.7.** *Let  $\beta_i > \alpha_i \geq 0$  and  $\alpha_{\check{i}} \geq 0$  for  $i = 1, 2$  and  $\check{i} \in \{1, 2\} \setminus \{i\}$ . Assume  $u_{\alpha_1, \alpha_2}, u_{\beta_1, \alpha_2}, u_{\alpha_1, \alpha_2}$ , and  $u_{\alpha_1, \beta_2}$  are any four minimizers of  $\mathcal{J}_{\alpha_1, \alpha_2}, \mathcal{J}_{\beta_1, \alpha_2}, \mathcal{J}_{\alpha_1, \alpha_2}$ , and  $\mathcal{J}_{\alpha_1, \beta_2}$ , respectively. Then*

$$\begin{aligned} \|T_1 u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Omega)} &\geq \|T_1 u_{\beta_1, \alpha_2} - g_1\|_{L^1(\Omega)} \quad \text{and} \\ \|T_2 u_{\alpha_1, \alpha_2} - g_2\|_{L^2(\Omega)} &\geq \|T_2 u_{\alpha_1, \beta_2} - g_2\|_{L^2(\Omega)} \end{aligned}$$

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

*Proof.* We start by showing the first inequality. Suppose it is not true, i.e.,  $\|T_1 u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Omega)} < \|T_1 u_{\beta_1, \alpha_2} - g_1\|_{L^1(\Omega)}$ . From the optimality of  $u_{\alpha_1, \alpha_2}$  we have that  $\mathcal{J}_{\alpha_1, \alpha_2}(u_{\alpha_1, \alpha_2}) \leq \mathcal{J}_{\alpha_1, \alpha_2}(u_{\beta_1, \alpha_2})$ . Then we have

$$\begin{aligned}\mathcal{J}_{\beta_1, \alpha_2}(u_{\alpha_1, \alpha_2}) &= \mathcal{J}_{\alpha_1, \alpha_2}(u_{\alpha_1, \alpha_2}) + (\beta_1 - \alpha_1) \|T_1 u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Omega)} \\ &\leq \mathcal{J}_{\alpha_1, \alpha_2}(u_{\beta_1, \alpha_2}) + (\beta_1 - \alpha_1) \|T_1 u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Omega)} \\ &< \mathcal{J}_{\alpha_1, \alpha_2}(u_{\beta_1, \alpha_2}) + (\beta_1 - \alpha_1) \|T_1 u_{\beta_1, \alpha_2} - g_1\|_{L^1(\Omega)} = \mathcal{J}_{\beta_1, \alpha_2}(u_{\beta_1, \alpha_2}),\end{aligned}$$

which is a contradiction, since  $u_{\beta_1, \alpha_2}$  is a minimizer of  $\mathcal{J}_{\beta_1, \alpha_2}$ . This proves the first inequality.

By similar arguments one can show again by contradiction that the second inequality holds.  $\square$

In order to show a stability result of the  $L^1$ - $L^2$ -TV model with respect to its parameters we adapt [9, Lemma 10.2] to our more general setting.

**Lemma 10.8.** *Let  $u \in BV(\Omega)$  be a minimizer of  $\mathcal{J}_{\alpha_1, \alpha_2}$ . Then for every  $v \in BV(\Omega)$  we have*

$$\alpha_2 \|T_2(u - v)\|_{L^2(\Omega)}^2 \leq \mathcal{J}_{\alpha_1, \alpha_2}(v) - \mathcal{J}_{\alpha_1, \alpha_2}(u).$$

*Proof.* By setting  $F(u) = \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + |Du|(\Omega)$  and  $G(u) = \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}$  the proof is analogue to the one of [9, Lemma 10.2].  $\square$

**Theorem 10.9.** *Define  $a_i(\alpha_1, \alpha_2) := \frac{1}{\alpha_i} (\alpha_1 \|g_1\|_{L^1(\Omega)} + \alpha_2 \|g_2\|_{L^2(\Omega)}^2)$  for  $i = 1, 2$  and let  $\alpha_1, \bar{\alpha}_1, \alpha_2, \bar{\alpha}_2 > 0$ . If  $u_{\alpha_1, \alpha_2}$  and  $u_{\bar{\alpha}_1, \bar{\alpha}_2}$  are minimizers of  $\mathcal{J}_{\alpha_1, \alpha_2}$  and  $\mathcal{J}_{\bar{\alpha}_1, \bar{\alpha}_2}$  respectively, then we have*

$$\|T_2(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)}^2 \leq \frac{|\alpha_1 - \bar{\alpha}_1|}{\alpha_2 + \bar{\alpha}_2} \max \{a_1(\bar{\alpha}_1, \bar{\alpha}_2), a_1(\alpha_1, \alpha_2)\} + \frac{|\alpha_2 - \bar{\alpha}_2|}{\alpha_2 + \bar{\alpha}_2} C =: B, \quad (10.10)$$

where  $C = \min\{C_1, C_2\}$  with  $C_1 = \max \{a_2(\bar{\alpha}_1, \bar{\alpha}_2), a_2(\alpha_1, \alpha_2)\}$ ,

$$C_2 = \frac{A_2^2 |\alpha_2 - \bar{\alpha}_2| + A_2 ((\alpha_2 - \bar{\alpha}_2)^2 A_2^2 + 4(\alpha_2 + \bar{\alpha}_2)|\alpha_1 - \bar{\alpha}_1| \max \{a_1(\bar{\alpha}_1, \bar{\alpha}_2), a_1(\alpha_1, \alpha_2)\})^{1/2}}{2(\alpha_2 + \bar{\alpha}_2)},$$

and  $A_2 := a_2(\bar{\alpha}_1, \bar{\alpha}_2)^{1/2} + a_2(\alpha_1, \alpha_2)^{1/2}$ .

If additionally  $T_1 = T_2 =: T$  then

$$\|T(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)} \leq \min\{\sqrt{B}, \tilde{B}\} \quad (10.11)$$

where  $\tilde{B} := \frac{|\alpha_1 - \bar{\alpha}_1|}{\alpha_2 + \bar{\alpha}_2} |\Omega|^{1/2} + \frac{|\alpha_2 - \bar{\alpha}_2|}{\alpha_2 + \bar{\alpha}_2} A_2$ .

*Proof.* By Lemma 10.8 we have

$$\begin{aligned}\alpha_2 \|T_2(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)}^2 &\leq \mathcal{J}_{\alpha_1, \alpha_2}(u_{\bar{\alpha}_1, \bar{\alpha}_2}) - \mathcal{J}_{\alpha_1, \alpha_2}(u_{\alpha_1, \alpha_2}) \\ \bar{\alpha}_2 \|T_2(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)}^2 &\leq \mathcal{J}_{\bar{\alpha}_1, \bar{\alpha}_2}(u_{\alpha_1, \alpha_2}) - \mathcal{J}_{\bar{\alpha}_1, \bar{\alpha}_2}(u_{\bar{\alpha}_1, \bar{\alpha}_2}).\end{aligned}$$

Summing up these inequalities yields

$$\begin{aligned}(\alpha_2 + \bar{\alpha}_2) \|T_2(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)}^2 &\leq (\alpha_1 - \bar{\alpha}_1) (\|T_1 u_{\bar{\alpha}_1, \bar{\alpha}_2} - g_1\|_{L^1(\Omega)} - \|T_1 u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Omega)}) \\ &\quad + (\alpha_2 - \bar{\alpha}_2) (\|T_2 u_{\bar{\alpha}_1, \bar{\alpha}_2} - g_2\|_{L^2(\Omega)}^2 - \|T_2 u_{\alpha_1, \alpha_2} - g_2\|_{L^2(\Omega)}^2).\end{aligned} \quad (10.12)$$

## II. Parameter Selection Methods for Total Variation Models

By the monotonicity property, see Proposition 10.7, we obtain that both terms on the right-hand side of the latter inequality are nonnegative. Moreover, note that

$$\|T_i u_{\alpha_1, \alpha_2} - g_i\|_{L^i(\Omega)}^i \leq a_i(\alpha_1, \alpha_2) \quad (10.13)$$

for  $i = 1, 2$  and any  $\alpha_1, \alpha_2 > 0$ , see Proposition 10.6. These observations lead to

$$\begin{aligned} \|T_2(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)}^2 &\leq \frac{|\alpha_1 - \bar{\alpha}_1|}{\alpha_2 + \bar{\alpha}_2} \max\{a_1(\bar{\alpha}_1, \bar{\alpha}_2), a_1(\alpha_1, \alpha_2)\} \\ &\quad + \frac{|\alpha_2 - \bar{\alpha}_2|}{\alpha_2 + \bar{\alpha}_2} \max\{a_2(\bar{\alpha}_1, \bar{\alpha}_2), a_2(\alpha_1, \alpha_2)\}. \end{aligned} \quad (10.14)$$

On the contrary, inequality (10.12) implies

$$\begin{aligned} (\alpha_2 + \bar{\alpha}_2) \|T_2(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)}^2 &\leq |\alpha_1 - \bar{\alpha}_1| \max\{a_1(\bar{\alpha}_1, \bar{\alpha}_2), a_1(\alpha_1, \alpha_2)\} \\ &\quad + |\alpha_2 - \bar{\alpha}_2| \|T_2(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)} (a_2(\bar{\alpha}_1, \bar{\alpha}_2)^{1/2} + a_2(\alpha_1, \alpha_2)^{1/2}) \end{aligned}$$

where we used the binomial formula  $a^2 - b^2 = (a+b)(a-b)$  for  $a, b \in \mathbb{R}$ , the triangle inequality, and (10.13). This is a quadratic inequality in  $\|T_2(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)}$  yielding

$$\begin{aligned} &\|T_2(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)} \\ &\leq \frac{|\alpha_2 - \bar{\alpha}_2|}{2(\alpha_2 + \bar{\alpha}_2)} A_2 + \frac{\sqrt{(\alpha_2 - \bar{\alpha}_2)^2 A_2^2 + 4(\alpha_2 + \bar{\alpha}_2)|\alpha_1 - \bar{\alpha}_1| \max\{a_1(\bar{\alpha}_1, \bar{\alpha}_2), a_1(\alpha_1, \alpha_2)\}}}{2(\alpha_2 + \bar{\alpha}_2)}. \end{aligned} \quad (10.15)$$

Squaring (10.15) and combining it with (10.14) yields the assertion.

If  $T_1 = T_2 = T$ , then from (10.12) by using the triangle inequality and the above used binomial formula we get

$$\begin{aligned} (\alpha_2 + \bar{\alpha}_2) \|T(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)}^2 &\leq |\alpha_1 - \bar{\alpha}_1| \|T(u_{\bar{\alpha}_1, \bar{\alpha}_2} - u_{\alpha_1, \alpha_2})\|_{L^1(\Omega)} \\ &\quad + (\alpha_2 - \bar{\alpha}_2) \|T(u_{\bar{\alpha}_1, \bar{\alpha}_2} - u_{\alpha_1, \alpha_2})\|_{L^2(\Omega)} (\|Tu_{\bar{\alpha}_1, \bar{\alpha}_2} - g_2\|_{L^2(\Omega)} + \|Tu_{\alpha_1, \alpha_2} - g_2\|_{L^2(\Omega)}). \end{aligned}$$

By using Hölder inequality on the  $L^1$ -term and by using (10.13) we obtain

$$\|T(u_{\alpha_1, \alpha_2} - u_{\bar{\alpha}_1, \bar{\alpha}_2})\|_{L^2(\Omega)} \leq \frac{|\alpha_1 - \bar{\alpha}_1|}{\alpha_2 + \bar{\alpha}_2} |\Omega|^{1/2} + \frac{|\alpha_2 - \bar{\alpha}_2|}{\alpha_2 + \bar{\alpha}_2} A_2.$$

Combining the latter inequality with (10.10) we get (10.11), which finishes the proof.  $\square$

**Remark 10.10.** If  $T_2 = I$ , then the inequalities (10.10) and (10.11) provide us with an upper bound on the distance between two solutions obtained with different parameters. In particular, if the parameters in the  $L^1$ - $L^2$ -TV model are slightly perturbed, only small changes are expected in the minimizer.

### 10.3.3. Further Properties of the $L^1$ - $L^2$ -TV Model

In this section we essentially follow [30] to further investigate and prove properties of the  $L^1$ - $L^2$ -TV model.

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

**Proposition 10.11.** *Given  $g_i \in L^2(\Omega)$ ,  $i = 1, 2$ . For each  $\alpha_1, \alpha_2 > 0$  we denote by  $u_{\alpha_1, \alpha_2}$  the unique minimizer of  $\mathcal{J}_{\alpha_1, \alpha_2}$  with  $T_1 = T_2 = I$ . Then the function  $(\alpha_1, \alpha_2) \rightarrow \alpha_1 \|u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Omega)} + \alpha_2 \|u_{\alpha_1, \alpha_2} - g_2\|_{L^2(\Omega)}^2$  is continuous.*

*Proof.* Fix  $\alpha_1^*, \alpha_2^* > 0$  and let  $u_{\alpha_1^*, \alpha_2^*}$  be the unique minimizer of  $\mathcal{J}_{\alpha_1^*, \alpha_2^*}$ . Let the sequence  $(\alpha_1^j, \alpha_2^j)_j$  converge to  $(\alpha_1^*, \alpha_2^*)$ . We consider the sequence  $(u_{\alpha_1^j, \alpha_2^j})_j$  of corresponding minimizers. From the relation  $\mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j}) \leq \mathcal{J}_{\alpha_1^j, \alpha_2^j}(0) = \alpha_1 \|g_1\|_{L^1(\Omega)} + \alpha_2 \|g_2\|_{L^2(\Omega)}^2$  follows that the sequence  $(u_{\alpha_1^j, \alpha_2^j})_j$  has uniformly bounded total variation,  $L^1$ -norm, and  $L^2$ -norm. Moreover, it implies that

$$\alpha_1 \|u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Omega)} + \alpha_2 \|u_{\alpha_1, \alpha_2} - g_2\|_{L^2(\Omega)}^2 \leq \alpha_1 \|g_1\|_{L^1(\Omega)} + \alpha_2 \|g_2\|_{L^2(\Omega)}^2. \quad (10.16)$$

The standard compactness property for functions with uniformly bounded total variation on compact sets implies that there exists a sequence, which we denote again by  $(u_{\alpha_1^j, \alpha_2^j})_j$ , such that  $u_{\alpha_1^j, \alpha_2^j} \rightarrow v \in L_{loc}^1(\Omega)$  in  $L^1$  on any bounded set. We may then pass to another subsequence to make sure that  $u_{\alpha_1^j, \alpha_2^j}(x) \rightarrow v(x)$  pointwise almost everywhere as well. Fatou's lemma shows that

$$\begin{aligned} \|v - g_2\|_{L^2(\Omega)} &\leq \liminf_{j \rightarrow \infty} \|u_{\alpha_1^j, \alpha_2^j} - g_2\|_{L^2(\Omega)} \\ \|v - g_1\|_{L^1(\Omega)} &\leq \liminf_{j \rightarrow \infty} \|u_{\alpha_1^j, \alpha_2^j} - g_1\|_{L^1(\Omega)} \end{aligned} \quad (10.17)$$

and hence  $v \in L^2(\Omega)$ . By the lower semicontinuity of the total variation, i.e.,  $|Dv|(\Omega) \leq \liminf_{j \rightarrow \infty} |Du_{\alpha_1^j, \alpha_2^j}|(\Omega)$ , we get  $\mathcal{J}_{\alpha_1^*, \alpha_2^*}(v) \leq \liminf_{j \rightarrow \infty} \mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j})$ .

Let us show that  $\mathcal{J}_{\alpha_1^*, \alpha_2^*}(u_{\alpha_1^*, \alpha_2^*}) \geq \limsup_{j \rightarrow \infty} \mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j})$ . Assume it is not true. Then there exists an  $\epsilon > 0$  and  $j$  arbitrary large such that  $\mathcal{J}_{\alpha_1^*, \alpha_2^*}(u_{\alpha_1^*, \alpha_2^*}) \leq \mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j}) - \epsilon$ . We also have  $\lim_{j \rightarrow \infty} \mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j}) = \mathcal{J}_{\alpha_1^*, \alpha_2^*}(u_{\alpha_1^*, \alpha_2^*})$  and hence  $\mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^*, \alpha_2^*}) < \mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j})$  for some large  $j$ , which is a contradiction, since  $u_{\alpha_1^j, \alpha_2^j}$  is a minimizer of  $\mathcal{J}_{\alpha_1^j, \alpha_2^j}$ . Hence we can conclude that

$$\limsup_{j \rightarrow \infty} \mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j}) \leq \mathcal{J}_{\alpha_1^*, \alpha_2^*}(u_{\alpha_1^*, \alpha_2^*}) \leq \mathcal{J}_{\alpha_1^*, \alpha_2^*}(v) \leq \liminf_{j \rightarrow \infty} \mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j}).$$

We thus see that  $v$  is a minimizer of  $\mathcal{J}_{\alpha_1^*, \alpha_2^*}$  and by uniqueness we get that  $v = u_{\alpha_1^*, \alpha_2^*}$ .

We are left by showing that

$$\limsup_{j \rightarrow \infty} \alpha_1^j \|u_{\alpha_1^j, \alpha_2^j} - g_1\|_{L^1(\Omega)} + \alpha_2^j \|u_{\alpha_1^j, \alpha_2^j} - g_2\|_{L^2(\Omega)}^2 \leq \alpha_1^* \|u_{\alpha_1^*, \alpha_2^*} - g_1\|_{L^1(\Omega)} + \alpha_2^* \|u_{\alpha_1^*, \alpha_2^*} - g_2\|_{L^2(\Omega)}^2.$$

Assume it is wrong. Then there exists an  $\epsilon > 0$  and arbitrary  $j$  such that

$$\alpha_1^j \|u_{\alpha_1^j, \alpha_2^j} - g_1\|_{L^1(\Omega)} + \alpha_2^j \|u_{\alpha_1^j, \alpha_2^j} - g_2\|_{L^2(\Omega)}^2 - \epsilon \geq \alpha_1^* \|u_{\alpha_1^*, \alpha_2^*} - g_1\|_{L^1(\Omega)} + \alpha_2^* \|u_{\alpha_1^*, \alpha_2^*} - g_2\|_{L^2(\Omega)}^2.$$

Then  $\mathcal{J}_{\alpha_1^*, \alpha_2^*}(u_{\alpha_1^*, \alpha_2^*}) \leq \liminf_{j \rightarrow \infty} \mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j}) - \epsilon$  and  $\mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^*, \alpha_2^*}) \rightarrow \mathcal{J}_{\alpha_1^*, \alpha_2^*}(u_{\alpha_1^*, \alpha_2^*})$  as  $j \rightarrow \infty$ . These last two statements lead as before to the contradiction that  $\mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^*, \alpha_2^*}) \leq \mathcal{J}_{\alpha_1^j, \alpha_2^j}(u_{\alpha_1^j, \alpha_2^j})$ . Hence we established continuity of the map for  $\alpha_1, \alpha_2 > 0$ .  $\square$

Now, let us deal with the behavior of the  $L^1$ - $L^2$ -TV model if the parameters  $\alpha_1$  and  $\alpha_2$  are small.

## II. Parameter Selection Methods for Total Variation Models

**Proposition 10.12.** *Let  $T_1 = T_2 = I$  and  $\alpha_1, \alpha_2 \geq 0$ . There exists a threshold  $\lambda^* = \lambda^*(\Omega)$  such that if  $\alpha_1|\Omega|^{\frac{1}{2}} + 2\alpha_2\|g_2\|_{L^2(\Omega)} < \lambda^*$ , then the minimizer  $u_{\alpha_1, \alpha_2}$  of  $\mathcal{J}_{\alpha_1, \alpha_2}$  is constant.*

*Proof.* Let  $u_\Omega := \frac{1}{|\Omega|} \int_\Omega u dx$ , then there exists a constant  $C > 0$  depending only on  $\Omega$  such that

$$|Du|(\Omega) \geq C\|u - u_\Omega\|_{L^2(\Omega)} \quad \text{for all } u \in BV(\Omega);$$

see [5, Remark 3.50] or [55, p. 24]. From the optimality of  $u := u_{\alpha_1, \alpha_2}$  we have  $\mathcal{J}_{\alpha_1, \alpha_2}(u) \leq \mathcal{J}_{\alpha_1, \alpha_2}(u_\Omega)$  and by the above inequality this yields

$$C\|u - u_\Omega\|_{L^2(\Omega)} + \alpha_1\|u - g_1\|_{L^1(\Omega)} + \alpha_2\|u - g_2\|_{L^2(\Omega)}^2 \leq \alpha_1\|u_\Omega - g_1\|_{L^1(\Omega)} + \alpha_2\|u_\Omega - g_2\|_{L^2(\Omega)}^2$$

which is equivalent to

$$C\|u - u_\Omega\|_{L^2(\Omega)} \leq \alpha_1\|u_\Omega - g_1\|_{L^1(\Omega)} - \alpha_1\|u - g_1\|_{L^1(\Omega)} + \alpha_2\|u_\Omega - g_2\|_{L^2(\Omega)}^2 - \alpha_2\|u - g_2\|_{L^2(\Omega)}^2.$$

By using the triangle inequality we get

$$C\|u - u_\Omega\|_{L^2(\Omega)} \leq \alpha_1\|u_\Omega - u\|_{L^1(\Omega)} + \alpha_2 \left( \|u_\Omega\|_{L^2(\Omega)}^2 - 2\langle u_\Omega - u, g_2 \rangle - \|u\|_{L^2(\Omega)}^2 \right).$$

Note, that by the Cauchy-Schwarz inequality  $\|u_\Omega\|_{L^2(\Omega)}^2 \leq \|u\|_{L^2(\Omega)}^2$  and hence we obtain

$$C\|u - u_\Omega\|_{L^2(\Omega)} \leq \alpha_1\|u_\Omega - u\|_{L^1(\Omega)} + 2\alpha_2\|u_\Omega - u\|_{L^2(\Omega)}\|g_2\|_{L^2(\Omega)}.$$

By using Hölder's inequality on the  $L^1$ -term we get

$$C\|u - u_\Omega\|_{L^2(\Omega)} \leq \left( \alpha_1|\Omega|^{\frac{1}{2}} + 2\alpha_2\|g_2\|_{L^2(\Omega)} \right) \|u - u_\Omega\|_{L^2(\Omega)}.$$

If  $C > \alpha_1|\Omega|^{\frac{1}{2}} + 2\alpha_2\|g_2\|_{L^2(\Omega)}$ , then  $\|u - u_\Omega\|_{L^2(\Omega)} = 0$  and hence  $u = u_\Omega$ , which shows the assertion with  $\lambda^* := C$ .  $\square$

A similar result is obtained for images  $g_1, g_2$  defined on  $\mathbb{R}^d$ ,  $d \in \mathbb{N}$ , where the variational problem is written as

$$\min_{u \in BV(\mathbb{R}^d)} \{ J_{\alpha_1, \alpha_2}(u) := \alpha_1\|u - g_1\|_{L^1(\mathbb{R}^d)} + \alpha_2\|u - g_2\|_{L^2(\mathbb{R}^d)}^2 + |Du|(\mathbb{R}^d) \}. \quad (10.18)$$

**Proposition 10.13.** *Let  $\Lambda \subset \mathbb{R}^d$  be a bounded domain,  $g_i \in L^i(\mathbb{R}^d)$  given such that  $\text{supp}(g_i) \subset \Lambda$  for  $i = 1, 2$ , and  $\alpha_1, \alpha_2 \geq 0$ . Then there exists a threshold  $\lambda^* = \lambda^*(\Lambda, d)$  such that if  $\alpha_1|\Lambda|^{\frac{1}{2}} + 2\alpha_2\|g_2\|_{L^2(\Lambda)} < \lambda^*$ , then a minimizer of  $J_{\alpha_1, \alpha_2}$  is given by  $u_{\alpha_1, \alpha_2} \equiv 0$ .*

*Proof.* By the Sobolev inequality, see e.g. [5, 55, 75], we have that there exists a constant  $C(d) > 0$  such that

$$\int_{\mathbb{R}^d} |Du| \geq C(d)\|u\|_{L^{\frac{d}{d-1}}(\mathbb{R}^d)} = C(d) \left( \|u\|_{L^{\frac{d}{d-1}}(\mathbb{R}^d \setminus \Lambda)}^{\frac{d}{d-1}} + \|u\|_{L^{\frac{d}{d-1}}(\Lambda)}^{\frac{d}{d-1}} \right)^{\frac{d-1}{d}}$$

for all  $u \in BV(\mathbb{R}^d)$  with compact support. Then from the minimality of  $u_{\alpha_1, \alpha_2}$  we have  $J_{\alpha_1, \alpha_2}(u_{\alpha_1, \alpha_2}) \leq J_{\alpha_1, \alpha_2}(0)$  and hence by the isoperimetric inequality this means

$$\begin{aligned} C(d)\|u_{\alpha_1, \alpha_2}\|_{L^{\frac{d}{d-1}}(\mathbb{R}^d)} + \alpha_1\|u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\mathbb{R}^d)} + \alpha_2\|u_{\alpha_1, \alpha_2} - g_2\|_{L^2(\mathbb{R}^d)}^2 \\ \leq \alpha_1\|g_1\|_{L^1(\mathbb{R}^d)} + \alpha_2\|g_2\|_{L^2(\mathbb{R}^d)}^2. \end{aligned} \quad (10.19)$$

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

Since  $\|\cdot\|_{L^i(\mathbb{R}^d)}^i = \|\cdot\|_{L^i(\mathbb{R}^d \setminus \Lambda)}^i + \|\cdot\|_{L^i(\Lambda)}^i$  and  $\text{supp}(g_i) \subset \Lambda$  for  $i = 1, 2$  we also have

$$C(d)\|u_{\alpha_1, \alpha_2}\|_{L^{\frac{d}{d-1}}(\Lambda)} + \alpha_1\|u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Lambda)} + \alpha_2\|u_{\alpha_1, \alpha_2} - g_2\|_{L^2(\Lambda)}^2 \leq \alpha_1\|g_1\|_{L^1(\Lambda)} + \alpha_2\|g_2\|_{L^2(\Lambda)}^2,$$

which is equivalent to

$$\begin{aligned} C(d)\|u_{\alpha_1, \alpha_2}\|_{L^{\frac{d}{d-1}}(\Lambda)} + \alpha_1\|u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Lambda)} + \alpha_2\|u_{\alpha_1, \alpha_2}\|_{L^2(\Lambda)}^2 + \alpha_2\|g_2\|_{L^2(\Lambda)}^2 \\ \leq \alpha_1\|g_1\|_{L^1(\Lambda)} + \alpha_2\|g_2\|_{L^2(\Lambda)}^2 + 2\alpha_2\langle u_{\alpha_1, \alpha_2}, g_2 \rangle. \end{aligned}$$

Now, we use the triangle inequality in the second term which yields

$$C(d)\|u_{\alpha_1, \alpha_2}\|_{L^{\frac{d}{d-1}}(\Lambda)} \leq \alpha_1\|u_{\alpha_1, \alpha_2}\|_{L^1(\Lambda)} + 2\alpha_2\langle u_{\alpha_1, \alpha_2}, g_2 \rangle.$$

In the latter inequality we multiply the left side by  $1 = \frac{1}{|\Lambda|^{\frac{2-d}{2d}}}\|1\|_{L^{\frac{2d}{2-d}}(\Lambda)}$  and use the generalized Hölder inequality, i.e.,  $\|uv\|_{L^r(\Lambda)} \leq \|u\|_{L^p(\Lambda)}\|v\|_{L^q(\Lambda)}$  for  $\frac{1}{p} + \frac{1}{q} = \frac{1}{r} \leq 1$  and  $u \in L^p(\Lambda)$ ,  $v \in L^q(\Lambda)$ , to get

$$\frac{C(d)}{|\Lambda|^{\frac{2-d}{2d}}}\|u_{\alpha_1, \alpha_2}\|_{L^2(\Lambda)} \leq \alpha_1\|u_{\alpha_1, \alpha_2}\|_{L^1(\Lambda)} + 2\alpha_2\|u_{\alpha_1, \alpha_2}\|_{L^2(\Lambda)}\|g_2\|_{L^2(\Lambda)},$$

where we used the Cauchy-Schwarz inequality on the right side. By using once more Hölder's inequality on the  $L^1$ -term we obtain

$$\frac{C(d)}{|\Lambda|^{\frac{2-d}{2d}}}\|u_{\alpha_1, \alpha_2}\|_{L^2(\Lambda)} \leq \left( \alpha_1|\Lambda|^{\frac{1}{2}} + 2\alpha_2\|g_2\|_{L^2(\Lambda)} \right) \|u_{\alpha_1, \alpha_2}\|_{L^2(\Lambda)}.$$

If  $\frac{C(d)}{|\Lambda|^{\frac{2-d}{2d}}} > \alpha_1|\Lambda|^{\frac{1}{2}} + 2\alpha_2\|g_2\|_{L^2(\Lambda)}$  then  $\|u_{\alpha_1, \alpha_2}\|_{L^2(\Lambda)} = 0$  and hence  $u_{\alpha_1, \alpha_2} = 0$  in  $\Lambda$ .

We are left with showing that  $u_{\alpha_1, \alpha_2} = 0$  in  $\mathbb{R}^d \setminus \Lambda$  if  $\|g_2\|_{L^2(\Lambda)} < \frac{1}{2\alpha_2} \left( \frac{C(d)}{|\Lambda|^{\frac{2-d}{2d}}} - \alpha_1|\Lambda|^{\frac{1}{2}} \right)$ .

By the inequality (10.19) we also have

$$C(d)\|u_{\alpha_1, \alpha_2}\|_{L^{\frac{d}{d-1}}(\mathbb{R}^d)} + \alpha_1\|u_{\alpha_1, \alpha_2} - g_1\|_{L^1(\Lambda)} + \alpha_2\|u_{\alpha_1, \alpha_2} - g_2\|_{L^2(\Lambda)}^2 \leq \alpha_1\|g_1\|_{L^1(\Lambda)} + \alpha_2\|g_2\|_{L^2(\Lambda)}^2.$$

Now we apply the triangle inequality and split the first term into integrations over  $\mathbb{R}^d \setminus \Lambda$  and  $\Lambda$ , which gives

$$\begin{aligned} C(d) \left( \|u_{\alpha_1, \alpha_2}\|_{L^{\frac{d}{d-1}}(\mathbb{R}^d \setminus \Lambda)}^{\frac{d}{d-1}} + \|u_{\alpha_1, \alpha_2}\|_{L^{\frac{d}{d-1}}(\Lambda)}^{\frac{d}{d-1}} \right)^{\frac{d-1}{d}} + \alpha_1(\|g_1\|_{L^1(\Lambda)} - \|u_{\alpha_1, \alpha_2}\|_{L^1(\Lambda)}) \\ + \alpha_2(\|g_2\|_{L^2(\Lambda)} - \|u_{\alpha_1, \alpha_2}\|_{L^2(\Lambda)})^2 \leq \alpha_1\|g_1\|_{L^1(\Lambda)} + \alpha_2\|g_2\|_{L^2(\Lambda)}^2. \end{aligned} \quad (10.20)$$

For  $\alpha_1|\Lambda|^{\frac{1}{2}} + 2\alpha_2\|g_2\|_{L^2(\Lambda)} < \frac{C(d)}{|\Lambda|^{\frac{2-d}{2d}}} =: \lambda^*$  we have that  $\|u_{\alpha_1, \alpha_2}\|_{L^p(\Lambda)} = 0$  for  $p \in [1, \infty]$  and hence we obtain by (10.20) that  $\|u_{\alpha_1, \alpha_2}\|_{L^{\frac{d}{d-1}}(\mathbb{R}^d \setminus \Lambda)} = 0$ , which concludes the proof.  $\square$

The assumptions  $\alpha_1|\Omega|^{\frac{1}{2}} + 2\alpha_2\|g_2\|_{L^2(\Omega)} < \lambda^*$  and  $\alpha_1|\Lambda|^{\frac{1}{2}} + 2\alpha_2\|g_2\|_{L^2(\Lambda)} < \lambda^*$  of the previous propositions clearly hold, if the parameters  $\alpha_1$  and  $\alpha_2$  are sufficiently small. These

## II. Parameter Selection Methods for Total Variation Models

results somehow merge the behavior of the  $L^1$ -TV and  $L^2$ -TV model for small parameters, cf. [30, 75].

The last two statements dealt with the behavior of the  $L^1$ - $L^2$ -TV model if  $\alpha_1$  and  $\alpha_2$  are small. Motivated by results for the  $L^1$ -TV model we state now properties of the  $L^1$ - $L^2$ -TV model if  $\alpha_1$  is large. In particular, as for the  $L^1$ -TV model, see [30, Lemma 5.5], we have the following statement:

**Lemma 10.14.** *Given  $g_1 = g_2 =: g \in BV(\mathbb{R}^d)$ ,  $d \in \mathbb{N}$ . Assume there exists a vector field  $\phi$  with the following properties:*

1.  $\phi(x) \in C_c^1(\mathbb{R}^d, \mathbb{R}^d)$ ,
2.  $|\phi(x)| \leq 1$  for all  $x \in \mathbb{R}^d$ ,
3.  $\int_{\mathbb{R}^d} g(x) \operatorname{div} \phi(x) dx = |Dg|(\mathbb{R}^d)$ .

*Then there exists a threshold  $\alpha_1^* \geq 0$  independent of  $\alpha_2$  such that the unique minimizer of  $J_{\alpha_1, \alpha_2}$  is given by  $u_{\alpha_1, \alpha_2} = g$  for all  $\alpha_1 \geq \alpha_1^*$  and  $\alpha_2 \geq 0$ .*

*Proof.* For any  $u \in BV(\mathbb{R}^d)$  we have

$$\begin{aligned} J_{\alpha_1, \alpha_2}(u) &= |Du|(\mathbb{R}^d) + \alpha_1 \int_{\mathbb{R}^d} |u - g| dx + \alpha_2 \int_{\mathbb{R}^d} |u - g|^2 dx \\ &\geq \int_{\mathbb{R}^d} u \operatorname{div} \phi dx + \alpha_1 \int_{\mathbb{R}^d} |u - g_1| dx + \alpha_2 \int_{\mathbb{R}^d} |u - g|^2 dx \\ &= \int_{\mathbb{R}^d} g \operatorname{div} \phi dx + \int_{\mathbb{R}^d} (u - g) \operatorname{div} \phi dx + \alpha_1 \int_{\mathbb{R}^d} |u - g| dx + \alpha_2 \int_{\mathbb{R}^d} |u - g|^2 dx \\ &= |Dg|(\mathbb{R}^d) + \int_{\mathbb{R}^d} (u - g) \operatorname{div} \phi dx + \alpha_1 \int_{\mathbb{R}^d} |u - g| dx + \alpha_2 \int_{\mathbb{R}^d} |u - g|^2 dx \\ &\geq J_{\alpha_1, \alpha_2}(g) + (\alpha_1 - \max_{x \in \mathbb{R}^d} |\operatorname{div} \phi|) \int_{\mathbb{R}^d} |u - g| dx + \alpha_2 \int_{\mathbb{R}^d} |u - g|^2 dx. \end{aligned}$$

For  $\alpha_1 \geq \alpha_1^* := \max_{x \in \mathbb{R}^d} |\operatorname{div} \phi|$ , the last inequality shows  $J_{\alpha_1, \alpha_2}(u) \geq J_{\alpha_1, \alpha_2}(g)$ . Assume  $u$  is a minimizer, which is unique, it follows that  $u \equiv g$ .  $\square$

When we apply Lemma 10.14 to binary images, we obtain the following theorem, cf. [30, Theorem 5.6].

**Theorem 10.15.** *Let  $\Lambda \subset \mathbb{R}^d$  be a bounded domain with  $C^2$  boundary. Let  $g(x) = g_1(x) = g_2(x) = 1_\Lambda(x)$  for all  $x \in \mathbb{R}^d$  and  $\alpha_2 \geq 0$ . Then there exists a threshold  $\alpha_1^* \geq 0$  such that whenever  $\alpha_1 > \alpha_1^*$ , the unique minimizer of  $J_{\alpha_1, \alpha_2}$  is  $g = 1_\Lambda$  itself.*

### 10.4. Automated Parameter Selection

In order to motivate the  $L^1$ - $L^2$ -TV model in [60] for the version (10.18) a simple and illustrative example is presented, in which its minimizer is compared with the one of the  $L^1$ -TV model, i.e., when  $\alpha_2 = 0$  in (10.18), and with the one of the  $L^2$ -TV model, i.e., when  $\alpha_1 = 0$  in (10.18). Note, that for  $\alpha_2 > 0$  the functional in (10.18) is strictly convex and hence has a unique minimizer. Moreover, if  $\alpha_1 = \alpha_2 = 0$ , then any constant function is a minimizer of problem (10.18).

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

The amazing fact we observe from [60, Example 2.1] is that the  $L^1$ - $L^2$ -TV model possesses the advantages of both other models, i.e., the  $L^1$ -TV model and  $L^2$ -TV model. That is, the  $L^1$ - $L^2$ -TV model is able to recover the original image, has a unique solution  $T_2 u_{\alpha_1, \alpha_2}$ , since it is strictly convex with respect to  $T_2 u$ , and preserves even smaller details than the  $L^2$ -TV model.

We recall that for  $g_1 = g_2 = 1_{B_r(0)}$  being the characteristic function of a disk  $B_r(0)$  centered at the origin with radius  $r > 0$  and  $T_1 = T_2 = I$ , the unique minimizer of (10.18) is given by

$$u_{\alpha_1, \alpha_2} = \begin{cases} 0 & \text{if } 0 \leq r < \frac{2}{2\alpha_2 + \alpha_1}, \\ \left(\frac{2\alpha_2 + \alpha_1}{2\alpha_2} - \frac{1}{\alpha_2 r}\right) 1_{B_r(0)} & \text{if } \frac{2}{2\alpha_2 + \alpha_1} \leq r \leq \frac{2}{\alpha_1}, \\ 1_{B_r(0)} & \text{if } r > \frac{2}{\alpha_1}. \end{cases} \quad (10.21)$$

From this we clearly see that for the  $L^1$ - $L^2$ -TV model there exist numerous different parameters  $\alpha_1$  and  $\alpha_2$  generating the same solution, even if  $\frac{2}{2\alpha_2 + \alpha_1} \leq r \leq \frac{2}{\alpha_1}$ .

Our parameter selection approach is motivated by Theorem 10.5, from which we know that if  $\alpha_i > 0$ , then indeed  $\|T_i u - g_i\|_{L^i(\Omega)}^i = \nu_i |\Omega|$  for this value of  $i \in \{1, 2\}$ . In order to formulate an algorithm based on (10.8) we assume that the feasible set  $U$  is nonempty.

### 10.4.1. Uzawa's Method

Assuming that  $\nu_1$  and  $\nu_2$  are at our disposal, we suggest to choose the parameters  $\alpha_1$  and  $\alpha_2$  depending on the constraints in (10.8). Hence the constrained minimization problem (10.8) might be solved by Uzawa's method [35]; see Algorithm 5 below with  $\nu_i(u^{(n)}) \equiv \nu_i$  constant. In general, as described in Section 10.2,  $\nu_1$  and  $\nu_2$  depend on the original (unknown) image. Nevertheless, instead of considering  $\nu_i(u)$  in (10.8), which would result in a quite nonlinear problem, we choose a reference image and compute approximate values  $\nu_1$  and  $\nu_2$ , leading to the following iterative scheme:

**Algorithm 5** (Uzawa's method). Initialize  $\rho > 0$  (small enough),  $\alpha_i^{(0)} > 0$  for  $i = 1, 2$  and set  $n = 0$ ;

- 1) Compute  $u^{(n)} \in \arg \min_{u \in BV(\Omega)} \mathcal{J}_{\alpha_1^{(n)}, \alpha_2^{(n)}}(u)$
- 2) Update  $\alpha_i^{(n+1)} = \max\{\alpha_i^{(n)} + \rho(H_i(u^{(n)}) - \nu_i(u^{(n)})|\Omega|), 0\}$  for  $i = 1, 2$ ;
- 3) Stop or set  $n = n + 1$  and continue with step 1).

Here and below,  $H_i(u) := \|T_i u - g_i\|_{L^i(\Omega)}^i$  and  $\nu_i(u^{(n)})$  is computed according to the formulas presented in Section 10.2 for  $i = 1, 2$ , i.e., (10.5) and (10.6). Observe, that if  $H_i(u^{(n)}) < \nu_i(u^{(n)})|\Omega|$ , then  $\alpha_i$  is decreased, which relaxes the corresponding constraint, while for  $H_i(u^{(n)}) > \nu_i(u^{(n)})|\Omega|$  the value  $\alpha_i$  is increased and hence the associated constraint enforced. In step 2) it is ensured that the parameters  $\alpha_i$  are always nonnegative, whereby they are allowed to reach 0. The algorithm is stopped as soon as one of the following two conditions hold for the first time:

(S1) the distance between  $H_i(u^{(n)})$  and  $\nu_i(u^{(n)})|\Omega|$  is sufficiently small, i.e.,  $\frac{|H_i(u^{(n)}) - \nu_i(u^{(n)})|\Omega||}{\nu_i(u^{(n)})|\Omega|} < \varepsilon_1 = 10^{-4}$ , or

## II. Parameter Selection Methods for Total Variation Models

(S2) the norm of the difference of two successive iterates  $\alpha_i^{(n)}$  and  $\alpha_i^{(n+1)}$  drops below a certain threshold, i.e.,  $\|\alpha_i^{(n)} - \alpha_i^{(n+1)}\| < \varepsilon_2 = 10^{-4}$ .

In order to obtain convergence at all, the parameter  $\rho > 0$  has to be chosen sufficiently small. If convergent, then clearly the magnitude of  $\rho$  has a significant influence on the convergence speed. In particular, a small  $\rho$  leads to a very slow convergence. Hence we would wish to choose  $\rho$  as large as possible but small enough such that the algorithm still converges. In all our numerical experiments we observed convergence of Algorithm 5 if we chose  $\rho$  at most 1. However, for this choice of  $\rho$  it turns out that the convergence speed is very slow (see Table 10.3 below), which makes this algorithm not really practical. Therefore, we present an alternative approach next.

### 10.4.2. The pAPS-Algorithm

In [68] a fully automated parameter selection algorithm for the  $L^1$ -TV model, i.e.,  $\alpha_2 = 0$  in (10.3), and the  $L^2$ -TV model, i.e.,  $\alpha_1 = 0$  in (10.3), is proposed. We recall, that in contrast to Uzawa's method in the algorithm from [68] no additional parameter has to be chosen to find the regularization parameter  $\alpha_1$  or  $\alpha_2$  such that  $u_{\alpha_1,0}$  solves

$$\min_{u \in BV(\Omega)} |Du|(\Omega) \quad \text{s.t.} \quad H_1(u) = \nu_1 |\Omega| \quad (10.22)$$

and  $u_{0,\alpha_2}$  is a minimizer of

$$\min_{u \in BV(\Omega)} |Du|(\Omega) \quad \text{s.t.} \quad H_2(u) = \nu_2 |\Omega|. \quad (10.23)$$

The automated adjustment of the regularization parameter ( $\alpha_1$  or  $\alpha_2$ ) is performed iteratively depending on the constraint  $H_1(u) = \nu_1 |\Omega|$  in the case of the  $L^1$ -TV model or on the constraint  $H_2(u) = \nu_2 |\Omega|$  in the case of the  $L^2$ -TV model. For example, for the  $L^1$ -TV model the parameter  $\alpha_1$  is increased whenever  $\frac{H_1(u_{\alpha_1,0})}{\nu_1 |\Omega|} > 1$  and decreased if  $\frac{H_1(u_{\alpha_1,0})}{\nu_1 |\Omega|} < 1$ . This leads to the following update scheme:

$$\alpha_1^{(n+1)} = \left( \frac{H_1(u_{\alpha_1^{(n)},0})}{\nu_1 |\Omega|} \right)^p \alpha_1^{(n)},$$

where  $p \geq 0$  such that  $(H_1(u_{\alpha_1^{(n)},0}))_n$  is monotonically decreasing, if  $H_1(u_{\alpha_1^{(0)},0}) > \nu_1 |\Omega|$ , and  $(H_1(u_{\alpha_1^{(n)},0}))_n$  is monotonically increasing, if  $H_1(u_{\alpha_1^{(0)},0}) \leq \nu_1 |\Omega|$ .

Motivated by this strategy, we suggest the following automated parameter selection algorithm for the  $L^1$ - $L^2$ -TV model.

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

**Algorithm 6** (pAPS-Algorithm). Initialize  $p > 0$ ,  $\alpha_i^{(0)} > 0$  for  $i = 1, 2$  and set  $n = 0$ ;

- 1) Compute  $u^{(n)} \in \arg \min_{u \in BV(\Omega)} \mathcal{J}_{\alpha_1^{(n)}, \alpha_2^{(n)}}(u)$
- 2) Update  $\alpha_i^{(n+1)} = \left( \frac{H_i(u^{(n)})}{\nu_i(u^{(n)})|\Omega|} \right)^p \alpha_i^{(n)}$  for  $i = 1, 2$ ;
- 3) Solve  $u^{(n+1)} \in \arg \min_{u \in BV(\Omega)} \mathcal{J}_{\alpha_1^{(n+1)}, \alpha_2^{(n+1)}}(u)$
- 4) For  $i = 1, 2$  do
  - (a) if  $H_i(u^{(0)}) \leq \nu_i(u^{(0)})|\Omega|$ 
    - (i) if  $H_i(u^{(n+1)}) > \nu_i(u^{(n+1)})|\Omega|$ , decrease  $p$ , e.g.,  $p = p/2$ , and go to step 2);
    - (ii) if  $H_i(u^{(n+1)}) \leq \nu_i(u^{(n+1)})|\Omega|$ , continue;
  - (b) if  $H_i(u^{(0)}) > \nu_i(u^{(0)})|\Omega|$ 
    - (i) if  $H_i(u^{(n+1)}) < \nu_i(u^{(n+1)})|\Omega|$ , decrease  $p$ , e.g.,  $p = p/2$ , and go to step 2);
    - (ii) if  $H_i(u^{(n+1)}) \geq \nu_i(u^{(n+1)})|\Omega|$ , continue;
- 5) Stop or set  $n := n + 1$  and return to step 2);

As a stopping criterion we use that either (S1), (S2), or

(S3) the power  $p$  is significant small, i.e.,  $p < \varepsilon_3 = 10^{-3}$ ;

holds for the first time.

Note, that  $(\nu_i(u^{(n)})|\Omega|)_n$ ,  $i = 1, 2$ , is in general not constant. Nevertheless, since for a mixture of noise the expected absolute value is here not available and difficult to compute, in our numerics we set  $\nu_1(\hat{u}) := \text{EAV}(\eta_{\hat{u}}) + \text{EAV}(\rho_{\hat{u}} | \hat{u})$ , which is actually only an above approximation of the real expected absolute value. However, note that  $\text{EAV}(\eta_{\hat{u}}) + \text{EAV}(\rho_{\hat{u}} | \hat{u})$  is also an element of  $[0, \sqrt{\frac{2}{\pi}}\sigma + \max\{s_1, s_2\}]$  or  $[0, \sqrt{\frac{2}{\pi}}\sigma + \frac{s}{2}]$ , respectively; cf. (10.7). If for example  $s_1 = s_2$ , as in the experiment of Figure 10.2, then by Section 10.2 we have  $\nu_1(u^{(n)}) = s_1 + \sqrt{\frac{2}{\pi}}\sigma$  (i.e.,  $\nu_1$  is independent on the image) and hence the sequence  $(\nu_1(u^{(n)})|\Omega|)_n$  is indeed constant, while  $\nu_2(u^{(n)}) = \sigma^2 + s_2(1 - Tu^{(n)})^2 + s_1(Tu^{(n)})^2 - (s_2 - (s_2 + s_1)Tu^{(n)})^2$  (i.e.,  $\nu_2$  is depending on the image), which allows the value  $\nu_2(u^{(n)})|\Omega|$  to change during the iterations, although the changes might be rather small; see Figure 10.2(c) and Figure 10.3(c).

Due to the adaptive choice of  $p$  in the pAPS-algorithm, we observe that the generated sequences  $(H_i(u^{(n)}))_n$  and  $(\alpha_i^{(n)})_n$  are monotonically decreasing or increasing, depending on the initial  $\alpha_i^{(0)}$ , for  $i = 1, 2$ , while for Algorithm 5 these monotonic behaviors are in general not guaranteed; see Figure 10.2, Figure 10.3 and Figure 10.4. In particular, we have the following result for the pAPS-algorithm.

**Lemma 10.16.** *The pAPS-algorithm generates monotone sequences  $(\alpha_i^{(n)})_n$ , for  $i = 1, 2$ . In particular, we have*

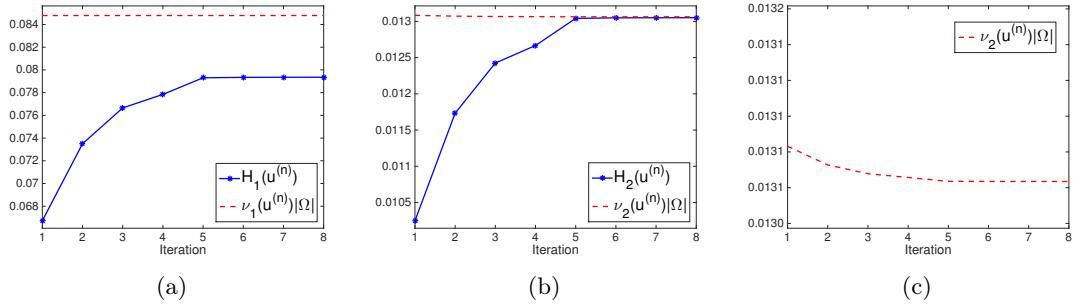
- (i) if  $\alpha_i^{(0)}$  is such that  $H_i(u^{(0)}) > \nu_i(u^{(0)})|\Omega|$ , then  $(\alpha_i^{(n)})_n$  is monotonically increasing, i.e.,  $\alpha_i^{(n)} \leq \alpha_i^{(n+1)}$  for all  $n \in \mathbb{N}$ ;

## II. Parameter Selection Methods for Total Variation Models

(ii) if  $\alpha_i^{(0)}$  is such that  $H_i(u^{(0)}) \leq \nu_i(u^{(0)})|\Omega|$ , then  $(\alpha_i^{(n)})_n$  is monotonically decreasing, i.e.,  $\alpha_i^{(n)} \geq \alpha_i^{(n+1)}$  for all  $n \in \mathbb{N}$ .

*Proof.* For  $H_i(u^{(0)}) > \nu_i(u^{(0)})|\Omega|$  we can show by induction that  $\alpha_i^{(n+1)} \geq \alpha_i^{(n)}$  for all  $n$  and  $i = 1, 2$ . In particular,  $H_i(u^{(n)}) \geq \nu_i(u^{(n)})|\Omega|$  implies  $\alpha_i^{(n+1)} = \left(\frac{H_i(u^{(n)})}{\nu_i(u^{(n)})|\Omega|}\right)^p \alpha_i^{(n)} \geq \alpha_i^{(n)}$ , where  $p$  is due to the pAPS-algorithm such that  $H_i(u^{(n+1)}) \geq \nu_i(u^{(n+1)})|\Omega|$ .

By similar arguments we obtain for  $\alpha_i^{(0)}$  with  $H_i(u^{(0)}) \leq \nu_i(u^{(0)})|\Omega|$ , that  $(\alpha_i^{(n)})_n$  is monotonically decreasing.  $\square$



**Figure 10.2:** Progress of  $H_i(u^{(n)})$  and  $\nu_i(u^{(n)})$ ,  $i = 1, 2$ , of the pAPS-algorithm with  $\alpha_1^{(0)} = 1 = \alpha_2^{(0)}$  for restoring the image ‘‘cameraman’’ (see Figure 10.5(a)) corrupted by Gaussian white noise with  $\sigma = 0.1$  and salt-and-pepper noise with  $s_1 = s_2 = 0.005$ .

Due to the monotonicity property of the sequence  $(\alpha_i^{(n)})_n$  for  $i = 1, 2$  we have the following convergence property of the pAPS-algorithm.

**Theorem 10.17.** For  $i \in \{1, 2\}$  the pAPS-algorithm generates a convergent sequence  $(\alpha_i^{(n)})_n$ , i.e.,  $\lim_{n \rightarrow \infty} \alpha_i^{(n)} = \bar{\alpha}_i \in \mathbb{R}$ , if one of the following conditions holds:

(i)  $\alpha_i^{(0)} > 0$  such that  $H_i(u^{(0)}) \leq \nu_i(u^{(0)})|\Omega|$ ;

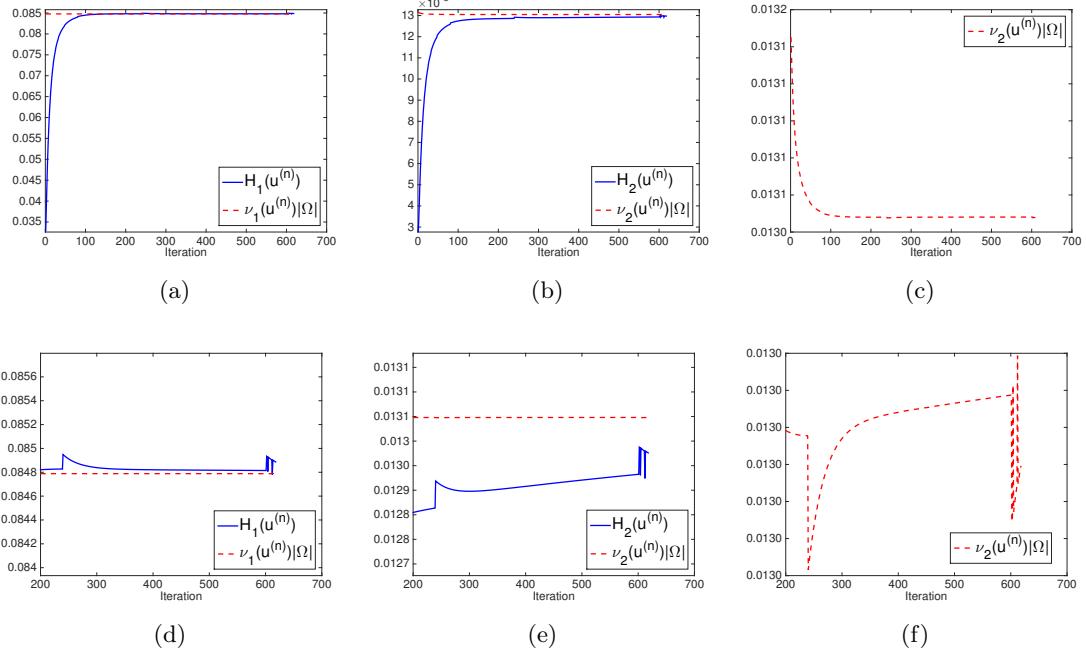
(ii) there exist  $\bar{\alpha}_1, \bar{\alpha}_2 > 0$  such that  $H_i(u^{(n)}) > \nu_i(u^{(n)})|\Omega|$  for all  $\alpha_i^{(n)} < \bar{\alpha}_i$  and  $H_i(u_{\alpha_1, \alpha_2}) \leq \nu_i(u_{\alpha_1, \alpha_2})|\Omega|$  for all  $\alpha_1 \geq \bar{\alpha}_1$  and  $\alpha_2 \geq \bar{\alpha}_2$ , where  $u_{\alpha_1, \alpha_2}$  is a solution of (10.3).

*Proof.* (i) Let  $\alpha_i^{(0)} > 0$  such that  $H_i(u^{(0)}) \leq \nu_i(u^{(0)})|\Omega|$ . Then by Lemma 10.16(ii) we have that  $(\alpha_i^{(n)})_n$  is monotonically decreasing, i.e.,  $0 \leq \alpha_i^{(n+1)} \leq \alpha_i^{(n)} \leq \alpha_i^{(0)}$  for all  $n \in \mathbb{N}$ , and hence it is bounded. The convergence follows by the monotone convergence theorem for sequences.

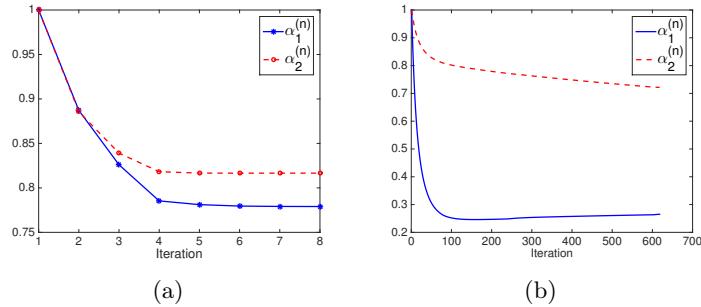
(ii) If there exist  $\bar{\alpha}_1, \bar{\alpha}_2 > 0$  such that  $H_i(u^{(n)}) > \nu_i(u^{(n)})|\Omega|$  for all  $\alpha_i^{(n)} < \bar{\alpha}_i$  and  $H_i(u_{\alpha_1, \alpha_2}) \leq \nu_i(u_{\alpha_1, \alpha_2})|\Omega|$  for all  $\alpha_1 \geq \bar{\alpha}_1$  and  $\alpha_2 \geq \bar{\alpha}_2$ , then  $(\alpha_i^{(n)})_n$  is monotonically increasing, cf. Lemma 10.16(i), and we deduce that  $0 \leq \alpha_i^{(n)} \leq \alpha_i^{(n+1)} < \bar{\alpha}_i$  for all  $n \in \mathbb{N}$ . Hence  $(\alpha_i^{(n)})_n$  is bounded and consequently convergent, which concludes the proof.  $\square$

Note, that  $\alpha_i^{(0)}$  has to be chosen positive for  $i = 1, 2$ , since if  $\alpha_i^{(0)} = 0$  in the pAPS-algorithm, then  $\alpha_i^{(n)} = 0$  for all  $n \geq 0$ , and we cannot expect a reasonable result in general.

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model



**Figure 10.3:** Progress of  $H_i(u^{(n)})$  and  $\nu_i(u^{(n)})$ ,  $i = 1, 2$ , of the Algorithm 5 with  $\alpha_1^{(0)} = 1 = \alpha_2^{(0)}$  for restoring the image “cameraman” (see Figure 10.5(a)) corrupted by Gaussian white noise with  $\sigma = 0.1$  and salt-and-pepper noise with  $s_1 = s_2 = 0.005$ . In (d) - (f) we zoomed in on the last few hundred iterations.



**Figure 10.4:** Progress of  $\alpha_1^{(n)}$  and  $\alpha_2^{(n)}$  of the pAPS-algorithm in (a) and Algorithm 5 in (b) with  $\alpha_1^{(0)} = 1 = \alpha_2^{(0)}$  for restoring the image “cameraman” (see Figure 10.5(a)) corrupted by Gaussian white noise with  $\sigma = 0.1$  and salt-and-pepper noise with  $s_1 = s_2 = 0.005$ .

## II. Parameter Selection Methods for Total Variation Models

### 10.4.3. Solving the Constrained Problem

Instead of determining the parameters  $\alpha_1$  and  $\alpha_2$  based on the constrained formulation (10.8) and then solving the  $L^1$ - $L^2$ -TV model with these parameters, as described above, we may alternatively compute a minimizer of the constrained optimization problem (10.8) directly. This can be done, for example, by using the alternating direction method (ADM) as in [78], where the ADM is applied for solving (10.22) and (10.23) in a finite dimensional setting. For the convenience of the reader we describe here a possible implementation of this approach adapted to our problem. Other strategies, which may be adjusted to solve problem (10.8), can be, for example, found in [22, 86, 90].

Similar as in [78] the ADM can be utilized to solve the constrained problem (10.8) in a finite dimensional setting, i.e.,

$$\min_{u \in \mathbb{R}^N} \|\nabla u\|_1 \quad \text{s.t.} \quad \frac{1}{N} \|T_i u - g_i\|_i^i \leq \nu_i \text{ for } i = 1, 2, \quad (10.24)$$

where  $N \in \mathbb{N}$  denotes the number of pixels in the image,  $g_i \in \mathbb{R}^N$  is the discrete observed data vector,  $T_i \in \mathbb{R}^{N \times N}$  denotes a discrete operator for  $i = 1, 2$ , and  $\nabla \in \mathbb{R}^{2N \times N}$  is the discrete gradient operator.  $\|\cdot\|_i$  refers to the standard definition of the  $\ell^i$ -norm, i.e.,  $\|u\|_i := \left(\sum_{j=1}^N |u_j|^i\right)^{\frac{1}{i}}$  and  $\langle \cdot, \cdot \rangle$  denotes the  $\ell^2$  inner product. Moreover,  $\hat{u} \in \mathbb{R}^N$  describes the original (unknown) data.

In order to apply the ADM to problem (10.24) we rewrite it as follows:

$$\min_{w \in \mathbb{R}^N \times \mathbb{R}^N} \|w\|_1 \quad \text{s.t.} \quad w = \nabla u, z_i = T_i u, \frac{1}{N} \|z_i - g_i\|_i^i \leq \nu_i \text{ for } i = 1, 2,$$

which is equivalent to

$$\min_{w \in \mathbb{R}^N \times \mathbb{R}^N, z_i \in \mathbb{R}^N} \|w\|_1 + \chi_{Z_1}(z_1) + \chi_{Z_2}(z_2) \quad \text{s.t.} \quad w = \nabla u, z_i = T_i u \text{ for } i = 1, 2,$$

where  $Z_i(\hat{u}) := \{z \in \mathbb{R}^N : \frac{1}{N} \|z - g_i\|_i^i \leq \nu_i(\hat{u})\}$  for  $i = 1, 2$  and  $\chi_Z$  is the characteristic function of the set  $Z$ , i.e.,  $\chi_Z(z) = \begin{cases} 0 & \text{if } z \in Z, \\ \infty & \text{otherwise.} \end{cases}$

The augmented Lagrangian of this problem is

$$\mathcal{L}(u, v, \lambda) = f(v; \hat{u}) + \langle \lambda, Bu - v \rangle + \frac{\beta}{2} \|Bu - v\|_2^2,$$

with  $v = \begin{pmatrix} w \\ z_1 \\ z_2 \end{pmatrix} \in \mathbb{R}^{4N}$ ,  $f(v; \hat{u}) = \|w\|_1 + \chi_{Z_1(\hat{u})}(z_1) + \chi_{Z_2(\hat{u})}(z_2)$ ,  $B = \begin{pmatrix} \nabla \\ T_1 \\ T_2 \end{pmatrix} \in \mathbb{R}^{4N \times N}$ , and

$\beta > 0$  denoting the penalty parameter. Hence the ADM for solving (10.24) runs as follows:

**Algorithm 7** (ADM). Initialize  $v^{(0)} \in \mathbb{R}^{4N}$ ,  $\lambda^{(0)} \in \mathbb{R}^{4N}$  and set  $n = 0$ ;

- 1) Compute  $u^{(n+1)} \in \arg \min_u \langle \lambda^{(n)}, Bu - v^{(n)} \rangle + \frac{\beta}{2} \|Bu - v^{(n)}\|_2^2$
- 2) Compute  $v^{(n+1)} = \arg \min_v f(v; u^{(n+1)}) + \langle \lambda^{(n)}, Bu^{(n+1)} - v \rangle + \frac{\beta}{2} \|Bu^{(n+1)} - v\|_2^2$
- 3) Update  $\lambda^{(n+1)} = \lambda^{(n)} + \beta(Bu^{(n+1)} - v^{(n+1)})$
- 4) Stop or set  $n = n + 1$  and continue with step 1).

In order to obtain  $u^{(n+1)}$  in step 1) a linear system that may be diagonalized by the DFT is

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

to solve. Since  $\hat{u}$  is not at our disposal, we use in the function  $f$  instead the current approximate  $u^{(n+1)}$ ; see step 2). Then the solution of the minimization problem in step 2) might be computed as described in [78, Section 4.2] by soft thresholding and projection onto a weighted  $\ell^i$ -ball,  $i = 1, 2$ ; see [89] for detailed information on how to implement such a projection.

Although, the ADM is convergent for any  $\beta > 0$ , see for example [17, 51, 56], in our implementation we adjust  $\beta$  in every iteration as proposed in [78] by starting with  $\beta^{(0)} = 100$  in order to obtain a good numerical performance. Moreover, we use the same stopping criterion as suggested in [78].

### 10.5. An Algorithm for Solving the $L^1$ - $L^2$ -TV Model

For computing a minimizer of problem (10.3) different strategies might be used, as a primal-dual method or alternating direction method [17, 51, 66] among others. For example, in [3] in a finite element setting the primal-dual algorithm of Chambolle and Pock [27] is used for solving the  $L^1$ - $L^2$ -TV model. However, it is not the scope of this paper to compare different algorithms in order to detect the most efficient one, although this is an interesting research topic in its own right. Here we consider the algorithm suggested in [60] (without any convergence analysis), which is an adaptation of a method that was originally proposed for  $L^1$ -TV minimization problems in [8], based on replacing the functional  $\mathcal{J}_{\alpha_1, \alpha_2}$  by

$$F(u, v) := \alpha_1 \|v\|_{L^1(\Omega)} + \frac{1}{2\gamma} \|T_1 u - g_1 - v\|_{L^2(\Omega)}^2 + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + |Du|(\Omega), \quad (10.25)$$

where  $\gamma > 0$  is small, so that we have  $g_1 \approx T_1 u - v$ . Actually for  $\gamma \rightarrow 0$  (10.25) approaches the objective functional in (10.3). Then (10.25) is minimized alternating with respect to  $u$  and  $v$  which results in the following algorithm:

**Algorithm 8.** Initialize  $u^{(0)} \in L^2(\Omega)$ . For  $n = 0, 1, \dots$  do

$$v^{(n+1)} = \arg \min_{v \in L^2(\Omega)} \alpha_1 \|v\|_{L^1(\Omega)} + \frac{1}{2\gamma} \|T_1 u^{(n)} - g_1 - v\|_{L^2(\Omega)}^2 \quad (10.26)$$

$$u^{(n+1)} \in \arg \min_{u \in L^2(\Omega)} \frac{1}{2\gamma} \|T_1 u - g_1 - v^{(n+1)}\|_{L^2(\Omega)}^2 + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + |Du|(\Omega) \quad (10.27)$$

Now we are going to analyse the convergence of this algorithm.

**Theorem 10.18.** Assume there exist  $i \in \{1, 2\}$  such that  $T_i$  does not annihilate constant functions and  $\alpha_i > 0$ . Then weak accumulation points of the sequence  $(u^{(n)}, v^{(n)})_n$  generated by Algorithm 8 are minimizers of  $F$  in  $L^2(\Omega) \times L^2(\Omega)$  and  $BV(\Omega) \times L^2(\Omega)$ .

The proof of this statement uses the same ideas as the ones of Proposition 5 in [8]. However, in contrary to [8] where the proof is done in a finite dimensional setting with the assumption of a continuous objective functional, we are working in an infinite dimensional space and our functional  $F$  is only lower semicontinuous, which requires additional arguments. Because of these reasons we state the complete proof here.

*Proof.* By Algorithm 8 we have

$$F(u^{(n)}, v^{(n)}) \geq F(u^{(n)}, v^{(n+1)}) \geq F(u^{(n+1)}, v^{(n+1)}). \quad (10.28)$$

## II. Parameter Selection Methods for Total Variation Models

Since  $F$  is bounded below by 0 it follows that  $(F(u^{(n)}, v^{(n)}))_n$  is convergent. Note that  $F$  is coercive in  $L^2(\Omega) \times L^2(\Omega)$ . From this and the convergence of  $(F(u^{(n)}, v^{(n)}))_n$  we deduce that  $(u^{(n)}, v^{(n)})_n$  is bounded in  $L^2(\Omega) \times L^2(\Omega)$  and hence we can extract a weakly convergent subsequence. Moreover, due to the presence of the total variation  $|Du|(\Omega)$  in  $F$  and  $\alpha_1 + \alpha_2 > 0$  we obtain that  $(u^{(n)}, v^{(n)})_n$  is bounded in  $BV(\Omega) \times L^2(\Omega)$ . The compact embedding  $BV(\Omega) \hookrightarrow L^q(\Omega)$ ,  $1 \leq q < \frac{d}{d-1}$  ( $d = 2$  is the dimension of  $\Omega$ ), implies that a subsequence  $(u^{(n_k)}, v^{(n_k)})_k$  converges in  $L^q(\Omega) \times L^2(\Omega)$  to a limit  $(u^*, v^*) \in L^2(\Omega) \times L^2(\Omega)$ . By [7, Prop. 10.1.1] we even have that  $(u^*, v^*) \in BV(\Omega) \times L^2(\Omega)$ ,  $\liminf_{n_k \rightarrow \infty} |Du^{(n_k)}|(\Omega) \geq |Du^*|(\Omega)$ , and  $(u^{(n_k)}, v^{(n_k)})_k$  weakly converges to  $(u^*, v^*)$  in  $BV(\Omega) \times L^2(\Omega)$  as  $n_k \rightarrow +\infty$ . Further, we have, for all  $n_k \in \mathbb{N}$

$$F(u^{(n_k)}, v^{(n_k+1)}) \leq F(u^{(n_k)}, v)$$

for all  $v \in L^2(\Omega)$  and

$$F(u^{(n_k)}, v^{(n_k)}) \leq F(u, v^{(n_k)}) \quad (10.29)$$

for all  $u \in L^2(\Omega)$ . Note that  $(v^{(n_k+1)})_k$  is again bounded and let us denote by  $\tilde{v}$  a corresponding cluster point.

Considering (10.28) we have that

$$F(u^{(n_k)}, v^{(n_k)}) - F(u^{(n_k+1)}, v^{(n_k+1)}) \geq F(u^{(n_k)}, v^{(n_k+1)}) - F(u^{(n_k+1)}, v^{(n_k+1)}).$$

Since  $F$  is bounded from below, we obtain  $\lim_{n_k \rightarrow \infty} [F(u^{(n_k)}, v^{(n_k)}) - F(u^{(n_k+1)}, v^{(n_k+1)})] = 0$  and consequently

$$0 = \lim_{n_k \rightarrow \infty} [F(u^{(n_k)}, v^{(n_k+1)}) - F(u^{(n_k+1)}, v^{(n_k+1)})] = F(u^*, \tilde{v}) - F(u^*, v^*). \quad (10.30)$$

By passing (10.26) to the limit we get that  $\tilde{v}$  is a solution of  $\min_{v \in L^2(\Omega)} \alpha_1 \|v\|_{L^1(\Omega)} + \frac{1}{2\gamma} \|T_1 u^* - g_1 - v\|_{L^2(\Omega)}^2$ . From (10.30) we know that  $F(u^*, \tilde{v}) = F(u^*, v^*)$  and hence

$$\alpha_1 \|\tilde{v}\|_{L^1(\Omega)} + \frac{1}{2\gamma} \|T_1 u^* - g_1 - \tilde{v}\|_{L^2(\Omega)}^2 = \alpha_1 \|v^*\|_{L^1(\Omega)} + \frac{1}{2\gamma} \|T_1 u^* - g_1 - v^*\|_{L^2(\Omega)}^2.$$

By the uniqueness of the solution ( $F(u^*, \cdot)$  is strictly convex) we conclude that  $\tilde{v} = v^*$ . Hence  $v^{(n_k+1)} \rightarrow v^*$  for  $n_k \rightarrow \infty$ .

Moreover,  $v^* = \arg \min_{v \in L^2(\Omega)} F(u^*, v)$ , i.e.

$$F(u^*, v^*) \leq F(u^*, v) \quad \text{for all } v \in L^2(\Omega). \quad (10.31)$$

And by passing (10.29) to the limit we obtain

$$F(u^*, v^*) \leq \left( \liminf F(u^{(n_k)}, v^{(n_k)}) \leq \liminf F(u, v^{(n_k)}) = \right) F(u, v^*) \quad \text{for all } u \in L^2(\Omega). \quad (10.32)$$

From the definition of  $F$  the inequality in (10.31) is equivalent to

$$0 \in \frac{1}{\gamma} (v^* - T_1 u^* + g_1) + \alpha_1 \partial \|v^*\|_{L^1(\Omega)} \quad (10.33)$$

and (10.32) is equivalent to

$$0 \in \frac{1}{\gamma} T_1^*(T_1 u^* - g_1 - v^*) + 2\alpha_2 T_2^*(T_2 u^* - g_2) + \partial |Du^*|(\Omega). \quad (10.34)$$

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

The subdifferential of  $F$  at  $(u^*, v^*)$  is given by

$$\partial F(u^*, v^*) = \left( \begin{array}{l} \frac{1}{\gamma} T_1^*(T_1 u^* - g_1 - v^*) + 2\alpha_2 T_2^*(T_2 u^* - g_2) + \partial|Du^*|(\Omega) \\ \frac{1}{\gamma}(v^* - T_1 u^* + g_1) + \alpha_1 \partial\|v^*\|_{L^1(\Omega)} \end{array} \right).$$

According to (10.33) and (10.34) we have

$$\left( \begin{array}{l} 0 \\ 0 \end{array} \right) \in \partial F(u^*, v^*)$$

which is equivalent to  $F(u^*, v^*) = \min_{(u,v) \in L^2(\Omega) \times L^2(\Omega)} F(u, v)$ .  $\square$

The minimizer  $v^{(n+1)}$  of (10.26) can be easily computed via a soft thresholding, i.e.,  $v^{(n+1)} = \text{ST}(T_1 u^{(n)} - g_1, \gamma \alpha_1)$ , where

$$\text{ST}(g, \beta)(x) = \begin{cases} g(x) - \beta & \text{if } g(x) > \beta, \\ 0 & \text{if } |g(x)| \leq \beta, \\ g(x) + \beta & \text{if } g(x) < -\beta \end{cases}$$

for all  $x \in \Omega$ .

The solution of the minimization problem in (10.27) can be realized by replacing  $F$  by a family of *surrogate functionals*

$$\begin{aligned} S(u, a, v) := & F(u, v) + \frac{1}{2\gamma} \left( \delta_1 \|u - a\|_{L^2(\Omega)}^2 - \|T_1(u - a)\|_{L^2(\Omega)}^2 \right) \\ & + \alpha_2 \left( \delta_2 \|u - a\|_{L^2(\Omega)}^2 - \|T_2(u - a)\|_{L^2(\Omega)}^2 \right) \end{aligned}$$

with  $a, u, v \in L^2(\Omega)$  and  $\delta_i > \|T_i\|^2$  for  $i = 1, 2$ . Note that

$$\min_{u \in L^2(\Omega)} S(u, a, v) \Leftrightarrow \min_{u \in L^2(\Omega)} \left\| u - \frac{\gamma}{\delta_1 + 2\alpha_2 \delta_2 \gamma} \left( \frac{1}{\gamma} z_1 + 2\alpha_2 z_2 \right) \right\|_{L^2(\Omega)}^2 + \frac{2\gamma}{\delta_1 + 2\alpha_2 \delta_2 \gamma} |Du|(\Omega). \quad (10.35)$$

where  $z_1 = z_1(a) = \delta_1 a + T_1^*(g_1 + v - T_1 a)$  and  $z_2 = z_2(a) = \delta_2 a + T_2^*(g_2 - T_2 a)$ ; cf. [60]. There exist several numerical methods for solving (10.35) efficiently; see for example [10, 13, 15, 19, 23, 27, 31, 36, 37, 38, 41, 42, 43, 46, 57, 59, 62, 77, 81] and references therein. This leads to the following algorithm:

**Algorithm 9.** Initialize:  $u^{(0,L)} \in L^2(\Omega)$ . For  $n = 0, 1, \dots$  do

$$\begin{aligned} v^{(n+1)} &= \arg \min_{v \in L^2(\Omega)} \alpha_1 \|v\|_{L^1(\Omega)} + \frac{1}{2\gamma} \|T_1 u^{(n,L)} - g_1 - v\|_{L^2(\Omega)}^2 \\ u^{(n+1,0)} &= u^{(n,L)} \\ u^{(n+1,\ell+1)} &= \arg \min_{u \in L^2(\Omega)} S(u, u^{(n+1,\ell)}, v^{(n+1)}), \quad \ell = 0, \dots, L-1. \end{aligned} \quad (10.36)$$

Note that we do prescribe a finite number  $L \in \mathbb{N}$  of inner iterations.

**Theorem 10.19.** Let the assumption of Theorem 10.18 be satisfied and assume  $\delta_i > \|T_i\|^2$  for  $i = 1, 2$ . Then weak accumulation points of the sequence  $(u^{(n,L)}, v^{(n)})_n$  generated by Algorithm 9 are minimizers of  $F$  in  $L^2(\Omega) \times L^2(\Omega)$  and  $BV(\Omega) \times L^2(\Omega)$ .

## II. Parameter Selection Methods for Total Variation Models

*Proof.* By Algorithm 9 we have

$$\begin{aligned} F(u^{(n,L)}, v^{(n)}) &\geq F(u^{(n,L)}, v^{(n+1)}) = S(u^{(n+1,0)}, u^{(n+1,0)}, v^{(n+1)}) \geq S(u^{(n+1,1)}, u^{(n+1,0)}, v^{(n+1)}) \\ &\geq S(u^{(n+1,1)}, u^{(n+1,1)}, v^{(n+1)}) \geq \dots \geq S(u^{(n+1,L)}, u^{(n+1,L)}, v^{(n+1)}) = F(u^{(n+1,L)}, v^{(n+1)}). \end{aligned}$$

By the same arguments as in Theorem 10.18 we obtain

$$F(u^*, v^*) \leq F(u^*, v) \quad \text{for all } v \in L^2(\Omega),$$

where  $(u^*, v^*) \in BV(\Omega) \times L^2(\Omega)$  is a limit of the subsequence  $(u^{(n_k,L)}, v^{(n_k)})_k$ .

Next we want to show that  $0 \in \partial F(u^*, v^*)$ . Therefore we analyse the surrogate iteration (10.36) in more details. By the monotonic decrease of  $F$  and  $S$  we have

$$\begin{aligned} F(u^{(n,L)}, v^{(n)}) - F(u^{(n+1,1)}, v^{(n+1)}) &\geq F(u^{(n+1,0)}, v^{(n+1)}) - F(u^{(n+1,1)}, v^{(n+1)}) \\ &\geq S(u^{(n+1,1)}, u^{(n+1,0)}, v^{(n+1)}) - S(u^{(n+1,1)}, u^{(n+1,1)}, v^{(n+1)}) \\ &= \frac{1}{2\gamma} \left( \delta_1 \|u^{(n+1,1)} - u^{(n+1,0)}\|_{L^2(\Omega)}^2 - \|T_1(u^{(n+1,1)} - u^{(n+1,0)})\|_{L^2(\Omega)}^2 \right) \\ &\quad + \alpha_2 \left( \delta_2 \|u^{(n+1,1)} - u^{(n+1,0)}\|_{L^2(\Omega)}^2 - \|T_2(u^{(n+1,1)} - u^{(n+1,0)})\|_{L^2(\Omega)}^2 \right) \\ &\geq \left( \frac{1}{2\gamma} C_1 + \alpha_2 C_2 \right) \|u^{(n+1,1)} - u^{(n+1,0)}\|_{L^2(\Omega)}^2, \end{aligned}$$

where  $C_i := (\delta_i - \|T_i\|^2) > 0$ . Moreover, we get

$$F(u^{(n+1,\ell)}, v^{(n+1)}) - F(u^{(n+1,\ell+1)}, v^{(n+1)}) \geq \left( \frac{1}{2\gamma} C_1 + \alpha_2 C_2 \right) \|u^{(n+1,\ell+1)} - u^{(n+1,\ell)}\|_{L^2(\Omega)}^2.$$

Hence, after  $L$  steps we conclude

$$F(u^{(n,L)}, v^{(n)}) - F(u^{(n+1,L)}, v^{(n+1)}) \geq \left( \frac{1}{2\gamma} C_1 + \alpha_2 C_2 \right) \sum_{\ell=0}^{L-1} \|u^{(n+1,\ell+1)} - u^{(n+1,\ell)}\|_{L^2(\Omega)}^2. \quad (10.37)$$

Since the sequence  $(F(u^{(n,L)}, v^{(n)}))_n$  is convergent we deduce from (10.37) that

$$\sum_{\ell=0}^{L-1} \|u^{(n+1,\ell+1)} - u^{(n+1,\ell)}\|_{L^2(\Omega)}^2 \rightarrow 0, \quad n \rightarrow \infty,$$

and hence

$$\lim_{n \rightarrow \infty} \|u^{(n+1,\ell+1)} - u^{(n+1,\ell)}\|_{L^2(\Omega)}^2 = 0$$

for all  $\ell \in \{0, \dots, L-1\}$ . Consequently, the sequences  $(u^{(n_k,L)})$  and  $(u^{(n_k,L-1)})$  have the same limit  $u^*$ .

By the optimality of  $u^{(n_k,L)}$  we have

$$\begin{aligned} 0 &\in \partial S(\cdot, u^{(n_k,L-1)}, v^{(n_k)})(u^{(n_k,L)}) \\ &= \partial F(\cdot, v^{(n_k)})(u^{(n_k,L)}) + \frac{1}{\gamma} \left( \delta_1 (u^{(n_k,L)} - u^{(n_k,L-1)}) - T_1^* T_1 (u^{(n_k,L)} - u^{(n_k,L-1)}) \right) \\ &\quad + 2\alpha_2 \left( \delta_2 (u^{(n_k,L)} - u^{(n_k,L-1)}) - T_2^* T_2 (u^{(n_k,L)} - u^{(n_k,L-1)}) \right) \end{aligned}$$

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

Then, by letting  $n_k \rightarrow \infty$  we obtain

$$0 \in \partial S(\cdot, u^*, v^*)(u^*) = \partial F(\cdot, v^*)(u^*).$$

The rest of the proof is analogous to the proof of Theorem 10.18.  $\square$

**Remark 10.20** (Denoising). *If  $T_1 = T_2 = I$ , then we do not need surrogate functionals and use Algorithm 8 directly, since the minimization problem in (10.27) is equivalent to*

$$\arg \min_{u \in L^2(\Omega)} \left\| u - \frac{\gamma}{1 + 2\alpha_2\gamma} \left( \frac{1}{\gamma}(g_1 + v) + 2\alpha_2 g_2 \right) \right\|_{L^2(\Omega)}^2 + \frac{2\gamma}{1 + 2\alpha_2\gamma} |Du|(\Omega)$$

and can be solved as (10.35) by one of the methods mentioned above.

### 10.6. Numerical Experiments

In this section we present several numerical experiments on image denoising and image deblurring to show the behavior of the proposed algorithm and their restoration potential. As a comparison for the different restoration qualities of the image we use the PSNR [16] (peak signal-to-noise ratio) given by

$$\text{PSNR} = 20 \log \frac{1}{\|\hat{u} - u^*\|},$$

where  $\hat{u}$  denotes the original image before any corruption and  $u^*$  the restored image, which is widely used as an image quality assessment measure, and the MSSIM [88] (mean structural similarity), which usually relates to perceived visual quality better than PSNR. In general, when comparing PSNR and MSSIM, large values indicate better reconstruction than small values.

The minimization problem in the pAPS-algorithm as well as in Algorithm 5 is solved approximately by Algorithm 8 or Algorithm 9, where  $\gamma = 10^{-2}$ . Moreover, the initial power  $p$  in the pAPS-algorithm is chosen to be 1 in all our experiments.

For our numerical studies we consider the images shown in Figure 10.5 of size  $256 \times 256$  pixels. We recall, that the image intensity range of all examples considered in this paper is  $[0, 1]$ .

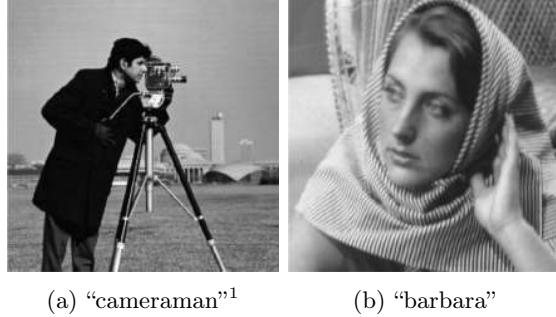
Since for a mixture of noise the expected absolute value is here not available and difficult to compute, in our numerics we set  $\nu_1(\hat{u}) := \text{EAV}(\eta_{\hat{u}}) + \text{EAV}(\rho_{\hat{u}} \mid \hat{u})$ , which is actually only an above approximation of the real expected absolute value. However, note that  $\text{EAV}(\eta_{\hat{u}}) + \text{EAV}(\rho_{\hat{u}} \mid \hat{u})$  is also an element of  $\left[0, \sqrt{\frac{2}{\pi}}\sigma + \max\{s_1, s_2\}\right]$  or  $\left[0, \sqrt{\frac{2}{\pi}}\sigma + \frac{s}{2}\right]$ , respectively. Moreover, we recall, that  $\nu_1$  as well as  $\nu_2$  are computed based on some approximation of the true image. More precisely, in Algorithm 5, the pAPS-algorithm, and the ADM  $\nu_i$  is updated in each iteration by  $\nu_i = \nu_i(u^{(n)})$ ,  $i = 1, 2$ .

All the presented experiments are performed in MATLAB on a MacBook Pro with 2.5 GHz Intel Core i7 processor.

---

<sup>1</sup>Used with kind permission, © Massachusetts Institute of Technology

## II. Parameter Selection Methods for Total Variation Models



**Figure 10.5:** Original images of size  $256 \times 256$  pixels.

$\alpha_1^{(0)}$	$\alpha_2^{(0)}$	PSNR	MSSIM	$\alpha_1^*$	$\alpha_2^*$
1	1	25.11	0.7163	0.7790	0.8166
1	0.5	25.16	0.7051	0.8541	0.4562
1	0.1	25.03	0.6998	0.9230	0.0971
0.5	1	24.87	0.7498	0.5241	1.2271
0.5	0.5	24.55	0.7478	0.5943	0.8918
0.5	0.1	24.36	0.7405	0.7175	0.3330
0.1	1	25.17	0.7470	0.1644	3.0531
0.1	0.5	25.14	0.7483	0.2150	2.7935
0.1	0.1	25.03	0.7509	0.3710	1.9926

**Table 10.1:** PSNR and MSSIM results for the  $256 \times 256$  pixel image “cameraman” corrupted by Gaussian white noise with  $\sigma = 0.1$  and salt-and-pepper noise with  $s_1 = s_2 = 0.005$  obtained by the pAPS-algorithm.

### 10.6.1. Initial Value $\alpha_i^{(0)}$

We start by investigating the pAPS-algorithm concerning its stability with respect to the initial  $\alpha_i^{(0)}$ ,  $i = 1, 2$ . For this purpose we consider the  $256 \times 256$  pixel image “cameraman” corrupted by Gaussian white noise with  $\sigma = 0.1$  and salt-and-pepper noise with  $s_1 = s_2 = 0.005$  and test for  $\alpha_1^{(0)}, \alpha_2^{(0)} \in \{0.1, 0.5, 1\}$ . Our findings are summarized in Table 10.1, i.e., the obtained parameters  $\alpha_1$  and  $\alpha_2$  and the PSNR and MSSIM of the corresponding received reconstructions. The obtained parameters  $\alpha_1$  and  $\alpha_2$  are always relatively close to the initial  $\alpha_1^{(0)}$  and  $\alpha_2^{(0)}$ . Note, that even if problem (10.8) may have a unique minimizer there may exist pairs  $(\alpha_1^1, \alpha_2^1)$  and  $(\alpha_1^2, \alpha_2^2)$  with  $(\alpha_1^1, \alpha_2^1) \neq (\alpha_1^2, \alpha_2^2)$  such that  $u_{\alpha_1^1, \alpha_2^1} = u_{\alpha_1^2, \alpha_2^2}$ , which can be, for example, easily seen from (10.21) and [60, Example 2.1]. We actually observe, that although the  $\alpha_1$ ’s and  $\alpha_2$ ’s differ significantly from each other, the PSNR and MSSIM seem similar throughout the experiments.

In order to keep the number of iterations in the pAPS-algorithm small a good choice of the initial values is still desirable. Therefore in the sequel we choose  $\alpha_i^{(0)}$ ,  $i = 1, 2$ , according to [73], i.e., we set  $\alpha_1^{(0)}$  and  $\alpha_2^{(0)}$  as in (10.4). By incorporating (10.4) for the choice of the initial parameters in the pAPS-algorithm makes this method fully automatic for the user.

### 10.6.2. Gaussian plus Impulse Noise

For the simultaneous removal of Gaussian and impulse noise we compare the performance of Algorithm 5, the pAPS-algorithm, and the ADM for solving directly the constrained problem (10.8) with the frequently used ROAD-trilateral filter [52], which is designed to remove a mixture of Gaussian noise (with zero mean and variance  $\sigma^2$ ) and impulse noise. This filter is based on a simple statistic to detect outliers in an image. Moreover, we also report on the results obtained by the  $L^1$ - $L^2$ -TV model with  $\alpha_1$  and  $\alpha_2$  chosen as suggested in [73], i.e., as in (10.4). In this case a minimizer is approximately computed by Algorithm 8 and in the case of deblurring by Algorithm 9. In the sequel we refer to them as the  $L^1$ - $L^2$ -TV algorithm. For our comparison we restore the ‘‘cameraman’’ image (see Figure 10.5(a)) and the ‘‘barbara’’ image (see Figure 10.5(b)) for mixed Gaussian-impulse noise with different noise levels, i.e.,  $\sigma \in \{0.01, 0.1, \sqrt{0.02}\}$ ,  $s_1 = s_2 \in \{0.005, 0.01, 0.05, 0.15\}$ , and  $s \in \{0.005, 0.01, 0.05, 0.15\}$ .

For simultaneously removing Gaussian and salt-and-pepper noise in the ‘‘cameraman’’ image we summarize our findings in Table 10.2. There it is demonstrated that the  $L^1$ - $L^2$ -TV algorithm with parameters chosen as in [73], i.e., as in equation (10.4), produces competitive results, which are actually always better than the ones generated by the ROAD-trilateral filter. Setting the initial parameters to (10.4) the pAPS-algorithm finds automatically new parameters  $(\alpha_1, \alpha_2)$  which improve the restoration quality of the  $L^1$ - $L^2$ -TV algorithm. In particular, in Figure 10.6 we see that the numerical solution produced by the  $L^1$ - $L^2$ -TV algorithm with the parameters as in [73] is over-smoothed, while the result generated by the pAPS-algorithm shows more details and has sharper edges. For this particular example in Figure 10.6 the ADM and Algorithm 5 produce the best results not only with respect to PSNR and MSSIM but also visually. Figure 10.6(d) and (e) show that edges are well preserved and noise is considerable removed. From Table 10.2 we further observe, that these two methods have the best performance with respect to PSNR and MSSIM when  $s_1 = s_2$  is sufficiently small. However, they show signs of weakness when  $s_1 = s_2$  is large, e.g.,  $s_1 = s_2 = 0.15$  in our experiments. In contrast, the pAPS-algorithm does not suffer from this weakness and outperforms the ADM and Algorithm 5 for these noise-levels. Moreover, the pAPS-algorithm gives always better MSSIM than the ROAD-trilateral filter and the  $L^1$ - $L^2$ -TV method with parameters as in (10.4); see Table 10.2 and Figure 10.6.

In Table 10.3 we report on the CPU-times (in seconds) for obtaining the results of Table 10.2. This table shows that Algorithm 5 converges tremendously slow and hence is in general not practical. The ADM needs about the same time for finding a minimizer of the constrained problem (10.8) as the pAPS-algorithm for generating a reasonable reconstruction. In the third and fourth column we also report on the CPU-times of the ROAD-trilateral filter as well as of the  $L^1$ - $L^2$ -TV algorithm, where no adjustment of the parameters is performed.

Almost similar observations as for denoising the ‘‘cameraman’’ image are also made for the ‘‘barbara’’ image; see Figure 10.7. We again see that the pAPS-algorithm produces a result which improves the one from the  $L^1$ - $L^2$ -TV algorithm with parameters as in [73]. Additionally now, for this image the pAPS-algorithm generates a reconstruction which is even better than the one of the ADM with respect to PSNR and MSSIM. When we look at Figure 10.7 then we observe that the ADM produces a slightly over-smoothed reconstruction. Hence the reason for the worse performance here might be this smoothing, which seems to do a good job for the ‘‘cameraman’’ image, see Figure 10.6, while for the ‘‘barbara’’ image, where for example the pattern on the scarf needs to be preserved, it is counterproductive leading to a smaller PSNR and MSSIM.

## II. Parameter Selection Methods for Total Variation Models

$\sigma$	$s_1 = s_2$	ROAD-trilateral		parameters as in [73]			ADM	
		PSNR	MSSIM	PSNR	MSSIM	$\alpha_1$	$\alpha_2$	PSNR
$\sqrt{0.02}$	0.005	22.42	0.6141	22.90	0.7116	0.2000	0.8000	24.92
	0.01	22.36	0.6134	23.01	0.7135	0.3333	0.6667	24.46
	0.05	21.86	0.5968	23.44	0.6619	0.7143	0.2857	22.09
	0.15	18.55	0.4543	21.96	0.5160	0.8824	0.1176	18.79
0.1	0.005	23.28	0.7011	23.29	0.7290	0.3333	0.6667	25.77
	0.01	23.25	0.7004	23.79	0.7419	0.5000	0.5000	25.25
	0.05	22.69	0.6859	24.27	0.6956	0.8333	0.1667	22.64
	0.15	20.28	0.5895	22.91	0.5954	0.9375	0.0625	18.96
0.01	0.005	24.78	0.8293	26.08	0.8625	0.9804	0.0196	33.09
	0.01	24.72	0.8283	26.02	0.8618	0.9901	0.0099	31.48
	0.05	23.99	0.8163	25.59	0.8509	0.9980	0.0020	27.18
	0.15	21.77	0.7562	24.27	0.8084	0.9993	0.0007	23.90

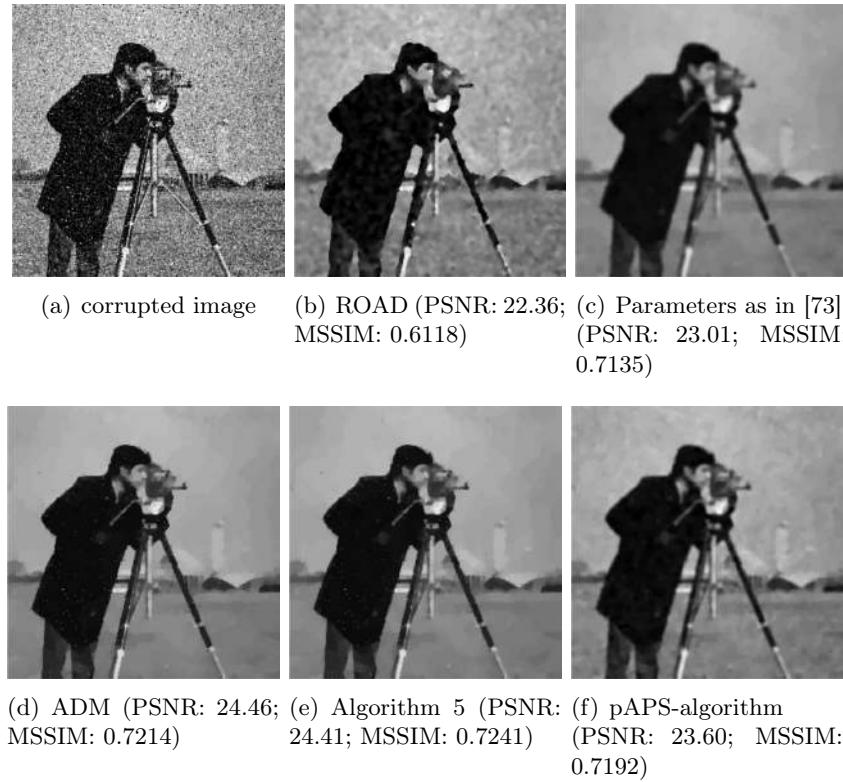
$\sigma$	$s_1 = s_2$	Algorithm 5				pAPS-algorithm			
		PSNR	MSSIM	$\alpha_1$	$\alpha_2$	PSNR	MSSIM	$\alpha_1$	$\alpha_2$
$\sqrt{0.02}$	0.005	24.85	0.7383	0.1171	0.8031	24.05	0.7245	0.2538	1.5717
	0.01	24.41	0.7241	0.1115	0.7250	23.60	0.7192	0.3720	1.0323
	0.05	22.12	0.6639	0	0.8923	22.86	0.6962	0.5114	0.2519
	0.15	18.76	0.5671	0	0.6951	21.04	0.6332	0.4392	0.0940
0.1	0.005	25.77	0.7515	0.2650	0.7215	24.95	0.7509	0.4552	1.5618
	0.01	25.36	0.7446	0.2871	0.5769	24.45	0.7429	0.5855	0.8821
	0.05	22.99	0.6870	0.1094	0.5808	23.80	0.7220	0.7013	0.1565
	0.15	18.96	0.5569	0	0.8207	22.59	0.6722	0.7002	0.0575
0.01	0.005	29.83	0.8561	1.0360	0	31.73	0.9461	2.6315	0.0373
	0.01	30.59	0.8822	1.0535	0	29.98	0.9375	2.1241	0.0143
	0.05	26.91	0.8375	0.8304	0	26.65	0.8844	1.3618	0.0021
	0.15	23.91	0.7558	0.5866	0	24.79	0.8206	1.2273	0.0008

**Table 10.2:** PSNR and MSSIM results for the  $256 \times 256$  pixel image ‘‘cameraman’’ corrupted by Gaussian white noise and salt-and-pepper noise. The parameters of the ROAD-trilateral filter are  $\sigma_S = 1$ ,  $\sigma_I = 40/255$ ,  $\sigma_J = 30/255$ , and  $\sigma_R$  is optimized between  $10/255$  and  $50/255$ , as suggested in [45].

$\sigma$	$s_1 = s_2$	ROAD	parameters as in [73]	ADM	Algorithm 5	pAPS-algorithm
$\sqrt{0.02}$	0.005	7	6	475	36950	290
	0.01	8	7	475	13323	293
	0.05	8	11	326	14227	362
	0.15	7	26	333	3081	515
0.1	0.005	7	6	506	54645	266
	0.01	7	7	420	94541	287
	0.05	7	12	406	17120	377
	0.15	7	25	329	10201	705
0.01	0.005	7	4	226	31090	88
	0.01	7	4	209	37974	95
	0.05	8	11	102	42191	288
	0.15	7	22	349	49262	1003

**Table 10.3:** CPU-times (in seconds) for obtaining the results of Table 10.2.

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

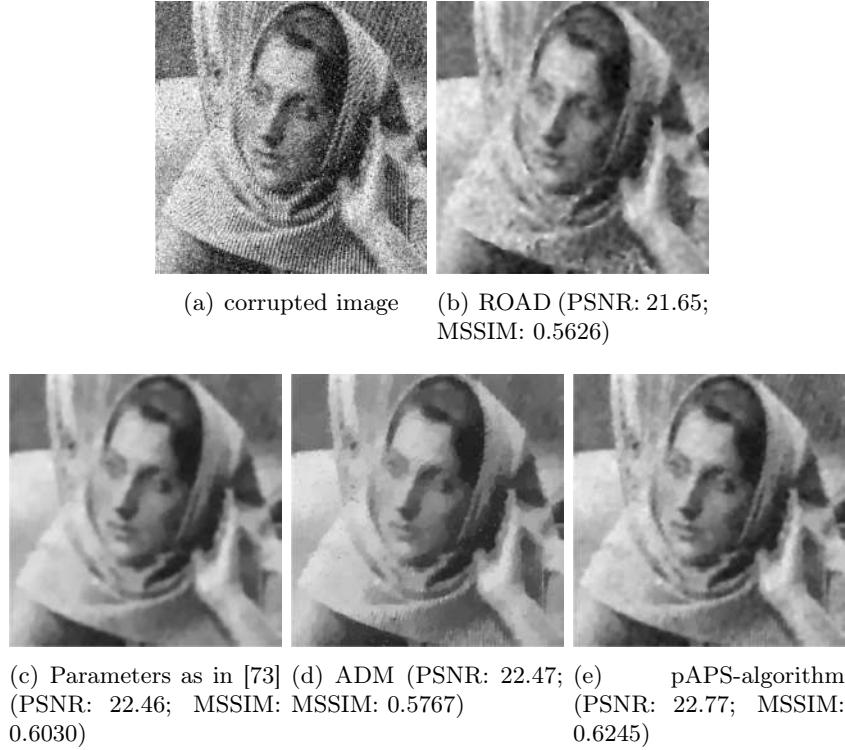


**Figure 10.6:** Reconstruction of the image “cameraman” corrupted by mixed Gaussian - salt-and-pepper noise with  $\sigma = \sqrt{0.02}$ ,  $s_1 = s_2 = 0.01$ .

For denoising the “cameraman” and “barbara” image corrupted by Gaussian and random-valued impulse noise we make analogues observations as above for removing a mixture of Gaussian and salt-and-pepper noise; see Table 10.4 and Figure 10.8. That is, as above the pAPS-algorithm improves the restoration quality of the  $L^1$ - $L^2$ -TV algorithm with parameters as in [73] or generates at least a result with the same MSSIM. This improvement is visible in Figure 10.8 for the “cameraman” image as well as for the “barbara” image. In particular, we see there for both examples that the results of the pAPS-algorithm preserve more details than the ones of the  $L^1$ - $L^2$ -TV algorithm with parameters as in [73]. For  $\sigma = \sqrt{0.02}$  and  $s = 0.01$  the ADM produces with respect to PSNR and MSSIM the best result for the “cameraman” image, while for the “barbara” image it is again topped by the pAPS-algorithm, see Figure 10.8. Also here we observe that the ROAD-trilateral filter is clearly outperformed by the  $L^1$ - $L^2$ -TV model.

Further we illustrate the successful application of our proposed algorithm when salt-and-pepper noise and Gaussian noise is disjoint present. More precisely, we consider the image in Figure 10.9 where the lower half  $g_1$  is contaminated with salt-and-pepper noise and in the upper half  $g_2$  only Gaussian noise is contained. This is an example where  $g_1 \neq g_2$ , although rather artificial, it is very interesting from a numerical point of view, since it is not possible to obtain a correct global solution by just cutting the image into an upper and a lower part due to the nonadditivity of the total variation [60]. Note, that, since  $g_1$  and  $g_2$  are disjoint and

## II. Parameter Selection Methods for Total Variation Models



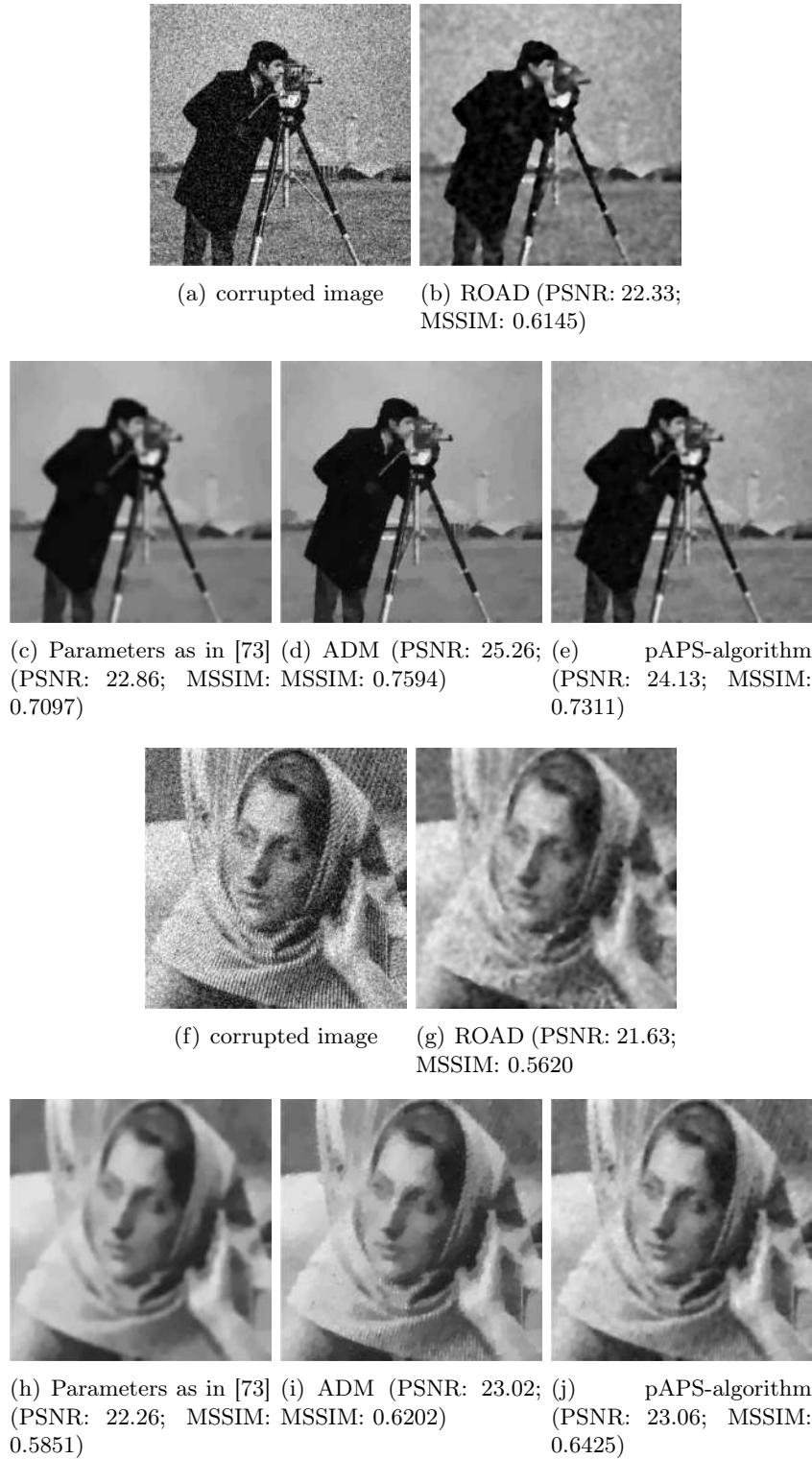
**Figure 10.7:** Reconstruction of the image “barbara” corrupted by mixed Gaussian - salt-and-pepper noise with  $\sigma = \sqrt{0.02}$ ,  $s_1 = s_2 = 0.01$ .

$T_1$  and  $T_2$  are restriction operators to the lower half and the upper half, respectively, problem (10.8) is superconsistent and consequently the feasible set  $U$  is nonempty. This justifies the use of the proposed pAPS-algorithm and the ADM for this setting. In particular, we demonstrate with the help of this algorithm that with the correct choice of the parameters  $\alpha_1$  and  $\alpha_2$  the  $L^1$ - $L^2$ -TV model is able to remove both types of noise considerably while preserving details at the same time from such images; see Figure 10.9(e). A similarly good result is obtained by solving the constrained problem (10.8) via the ADM; see Figure 10.9(d). On the contrary, the parameters according to (10.4) obviously yield an over-smoothed restoration, see Figure 10.9(c), and thus this parameter choice rule is not suitable for such an application. Figure 10.9 also shows that the ROAD-trilateral filter does not work well for this task, since it does not remove the Gaussian noise sufficiently and over-smooths the salt-and-pepper contaminated part.

### 10.6.3. Reconstruction of Blurred and Noisy Images

Now we are investigating the behavior of the proposed pAPS-algorithm and the ADM by solving (10.8) directly for reconstructing blurred images which are additionally contaminated by mixed noise. In particular, we consider again the “cameraman” and “barbara” image and add Gaussian blur with kernel size  $5 \times 5$  pixels and standard deviation 10 and corrupt it additionally by mixed Gaussian-impulse noise with  $\sigma = 15/255$ ,  $s_1 = s_2 = 0.01$  in the case of salt-and-pepper noise, and  $s = 0.01$  in the case of random-valued impulse noise. For the

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model



**Figure 10.8:** Reconstruction of the images “cameraman” and “barbara” corrupted by mixed Gaussian - random-valued impulse noise with  $\sigma = \sqrt{0.02}$ ,  $s = 0.01$ .

## II. Parameter Selection Methods for Total Variation Models

$\sigma$	$p$	ROAD-trilateral		parameters as in [73]			
		PSNR	MSSIM	PSNR	MSSIM	$\alpha_1$	$\alpha_2$
$\sqrt{0.02}$	0.005	22.34	0.5629	22.79	0.7071	0.0769	0.9231
	0.01	22.33	0.5620	22.86	0.7097	0.1429	0.8571
	0.05	22.23	0.5577	23.21	0.7104	0.4545	0.5455
	0.15	21.93	0.5439	22.89	0.6424	0.7143	0.2857
0.1	0.005	22.92	0.6060	23.00	0.7161	0.1429	0.8571
	0.01	22.93	0.6054	23.17	0.7232	0.2500	0.7500
	0.05	22.84	0.6028	24.03	0.7365	0.6250	0.3750
	0.15	22.65	0.5952	23.68	0.6823	0.8333	0.1667
0.01	0.005	23.26	0.6542	26.03	0.8600	0.9434	0.0566
	0.01	23.27	0.6540	26.08	0.8619	0.9709	0.0291
	0.05	23.21	0.6535	25.90	0.8578	0.9940	0.0060
	0.15	23.12	0.6518	25.32	0.8397	0.9980	0.0020

$\sigma$	$p$	ADM		pAPS-algorithm			
		PSNR	MSSIM	PSNR	MSSIM	$\alpha_1$	$\alpha_2$
$\sqrt{0.02}$	0.005	25.51	0.7669	24.30	0.7298	0.1076	2.2475
	0.01	25.26	0.7594	24.13	0.7311	0.1854	1.7897
	0.05	23.90	0.7152	23.21	0.7104	0.4545	0.5455
	0.15	21.78	0.6539	22.38	0.6721	0.5329	0.2592
0.1	0.005	26.78	0.7879	25.37	0.7633	0.2278	2.9287
	0.01	26.42	0.7787	25.11	0.7601	0.3602	2.0526
	0.05	24.66	0.7172	24.03	0.7365	0.6250	0.3750
	0.15	22.20	0.6446	23.59	0.6916	0.7666	0.1640
0.01	0.005	34.26	0.9400	33.57	0.9516	3.4075	0.3228
	0.01	33.41	0.9416	31.91	0.9480	2.6103	0.0790
	0.05	28.53	0.8951	28.88	0.9209	1.7779	0.0078
	0.15	26.69	0.8582	26.85	0.8734	1.5146	0.0024

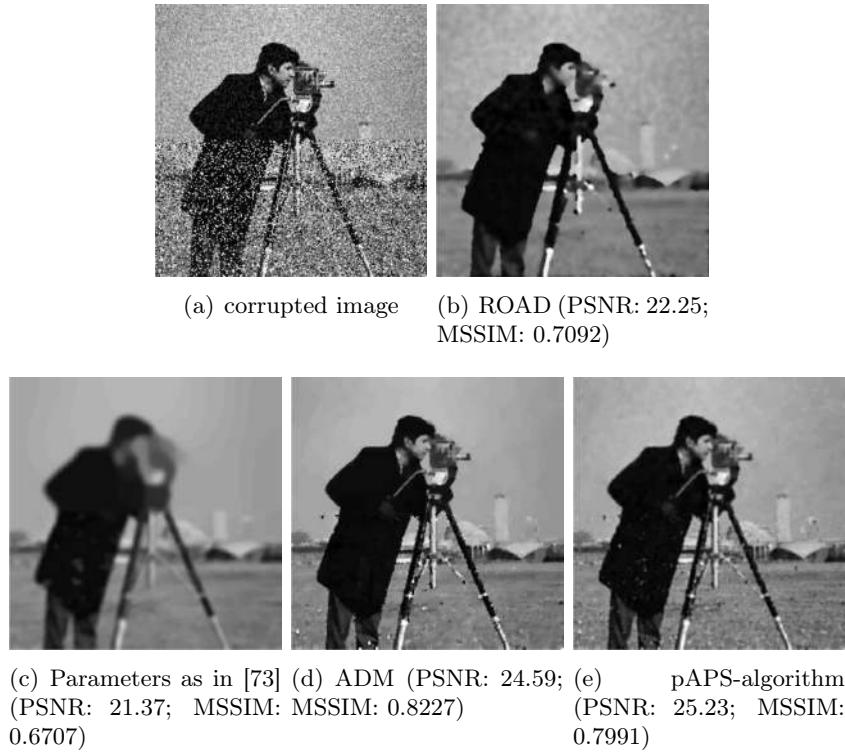
**Table 10.4:** PSNR and MSSIM results for the image ‘‘cameraman’’ corrupted by Gaussian white noise and random-valued impulse noise. The parameters of the ROAD-trilateral filter are  $\sigma_S = 1$ ,  $\sigma_I = 40/255$ ,  $\sigma_J = 30/255$ , and  $\sigma_R$  is optimized between 10/255 and 50/255, as suggested in [45].

sake of performance reference here we also compare the results of the pAPS-algorithm and the ADM with the ones obtained by the  $L^1$ - $L^2$ -TV algorithm with parameters as suggested in [73]. In Figure 10.10 and Figure 10.11 we show the respective results. We observe again that the parameters chosen by the pAPS-algorithm are more optimal than the ones suggested in [73], indicated by a larger PSNR and MSSIM. This is also visible in Figure 10.10 and Figure 10.11, where the reconstructions of the pAPS-algorithm seem less blurred. Moreover, as above with respect to PSNR and MSSIM the ADM generates a better reconstruction than the pAPS-algorithm for the ‘‘cameraman’’ image, while for the ‘‘barbara’’ image the pAPS performs best.

## 10.7. Conclusion

To fully utilize the strength and advantages of the  $L^1$ - $L^2$ -TV model the proper choice of the parameters  $\alpha_1$  and  $\alpha_2$  is essential, since they have a big influence on the restoration quality, as we see from our experiments. Therefore we present a fully automated algorithm, called pAPS-algorithm, for choosing appropriate parameters for the  $L^1$ - $L^2$ -TV minimization problem. The automated adjustment of the parameters is based on the discrepancy principle and inspired by the work in [68]. As initial values  $\alpha_i^{(0)}$  in the pAPS-algorithm we suggest to use the choice in (10.4). In this setting this method is fully automatic and generates parameters that give a

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model



**Figure 10.9:** Reconstruction of the image “cameraman” corrupted by Gaussian noise with  $\sigma = 0.1$  (upper part) and salt-and-pepper noise with  $s_1 = s_2 = 0.15$  (lower part).

satisfactory reconstruction, which are better than the ones obtained by the parameter-choice rule suggest in [73].

Due to the proposed automated parameter selection rule we are able to confirm and demonstrate once more that the  $L^1$ - $L^2$ -TV model is suitable to reconstruct images corrupted by mixed Gaussian-impulse noise and possibly some blur; cf. [60].

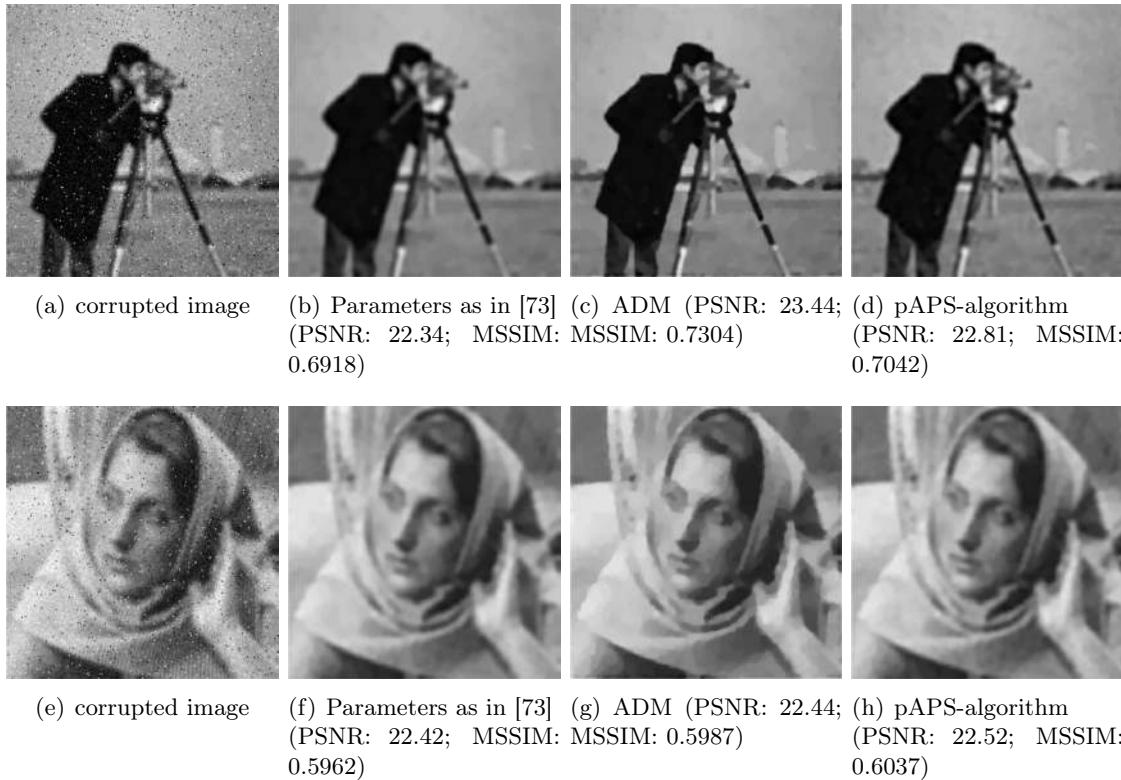
Moreover, we link the  $L^1$ - $L^2$ -TV model to the constrained optimization problem (10.8), which may be directly solved utilizing the ADM.

Future improvements of the  $L^1$ - $L^2$ -TV model may include spatially varying parameters, as considered in [48, 61, 68] for the  $L^1$ -TV model and  $L^2$ -TV model. In particular, large parameters  $\alpha_1$  and  $\alpha_2$  should perform well in regions with small texture, while small parameters remove noise considerable in homogeneous regions. Also including an impulse noise detector in the model might be of future interest to enhance its performance of removing mixed Gaussian-impulse noise.

## 10.8. References

- [1] R. Acar and C. R. Vogel. Analysis of bounded variation penalty methods for ill-posed problems. *Inverse Problems*, 10(6):1217–1229, 1994.
- [2] R. A. Adams and J. J. F. Fournier. *Sobolev Spaces*, volume 140 of *Pure and Applied Mathematics (Amsterdam)*. Elsevier/Academic Press, Amsterdam, second edition, 2003.

## II. Parameter Selection Methods for Total Variation Models



**Figure 10.10:** Reconstruction of the images ‘‘cameraman’’ and ‘‘barbara’’ image corrupted by Gaussian blur (kernel-size  $5 \times 5$ ; standard deviation 10) and mixed Gaussian - salt-and-pepper noise with  $\sigma = 15/255$ ,  $s_1 = s_2 = 0.01$ .

- [3] M. Alkämper and A. Langer. Using DUNE-ACFem for non-smooth minimization of bounded variation functions. *accepted by Archive of Numerical Software*, pages 1–13, 2016.
  - [4] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE Transactions on Signal Processing*, 45(4):913–917, 1997.
  - [5] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, 2000.
  - [6] A. Y. Aravkin, J. V. Burke, and M. P. Friedlander. Variational properties of value functions. *SIAM Journal on Optimization*, 23(3):1689–1717, 2013.
  - [7] H. Attouch, G. Buttazzo, and G. Michaille. *Variational Analysis in Sobolev and BV Spaces*. MOS-SIAM Series on Optimization. SIAM, Philadelphia, PA, 2014.
  - [8] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition – modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67(1):111–136, 2006.
  - [9] S. Bartels. *Numerical Methods for Nonlinear Partial Differential Equations*, volume 14. Springer, 2015.
  - [10] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18(11):2419–2434, 2009.

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model



**Figure 10.11:** Reconstruction of the “cameraman” and “barbara” image corrupted by Gaussian blur (kernel-size  $5 \times 5$ ; standard deviation 10) and mixed Gaussian - random-valued impulse noise with  $\sigma = 15/255$ ,  $s = 0.01$ .

- [11] F. Benvenuto, A. La Camera, C. Theys, A. Ferrari, H. Lantéri, and M. Bertero. The study of an iterative method for the reconstruction of images corrupted by poisson and gaussian noise. *Inverse Problems*, 24(3):035016, 2008.
- [12] M. Bertalmío, V. Caselles, B. Rougé, and A. Solé. TV based image restoration with local constraints. *Journal of Scientific Computing*, 19(1-3):95–122, 2003.
- [13] R. I. Bot and C. Hendrich. A Douglas–Rachford type primal-dual method for solving inclusions with mixtures of composite and parallel-sum type monotone operators. *SIAM Journal on Optimization*, 23(4):2541–2565, 2013.
- [14] S. Bonettini and V. Ruggiero. On the convergence of primal–dual hybrid gradient algorithms for total variation image restoration. *Journal of Mathematical Imaging and Vision*, 44(3):236–253, 2012.
- [15] R. I. Bot and C. Hendrich. Convergence analysis for a primal-dual monotone+ skew splitting algorithm with applications to total variation minimization. *Journal of Mathematical Imaging and Vision*, 49(3):551–568, 2014.
- [16] A. C. Bovik. *Handbook of Image and Video Processing*. Academic press, 2010.
- [17] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

## II. Parameter Selection Methods for Total Variation Models

- [18] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.
- [19] M. Burger, A. Sawatzky, and G. Steidl. First order algorithms in variational image processing. *arXiv preprint arXiv:1412.4237*, 2014.
- [20] J.-F. Cai, R. H. Chan, and M. Nikolova. Two-phase approach for deblurring images corrupted by impulse plus Gaussian noise. *Inverse Problems and Imaging*, 2(2):187–204, 2008.
- [21] L. Calatroni, C. Chung, J. C. D. L. Reyes, C.-B. Schönlieb, and T. Valkonen. Bilevel approaches for learning of variational imaging models. *arXiv preprint arXiv:1505.02120*, 2015.
- [22] M. Carlavan and L. Blanc-Féraud. Two constrained formulations for deblurring Poisson noisy images. In *Proceedings of the 18th IEEE International Conference on Image Processing (ICIP 2011)*, pages 689–692. IEEE, 2011.
- [23] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97, 2004.
- [24] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock. An introduction to total variation for image analysis. *Theoretical foundations and numerical methods for sparse recovery*, 9:263–340, 2010.
- [25] A. Chambolle and J. Darbon. On total variation minimization and surface evolution using parametric maximum flows. *International Journal of Computer Vision*, 84(3):288–307, 2009.
- [26] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, 1997.
- [27] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [28] R. H. Chan, Y. Dong, and M. Hintermüller. An efficient two-phase-TV method for restoring blurred images with impulse noise. *IEEE Transactions on Image Processing*, 19(7):1731–1739, 2010.
- [29] R. H. Chan, C.-W. Ho, and M. Nikolova. Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. *IEEE Transactions on Image Processing*, 14(10):1479–1485, 2005.
- [30] T. F. Chan and S. Esedoḡlu. Aspects of total variation regularized  $L^1$  function approximation. *SIAM Journal on Applied Mathematics*, 65(5):1817–1837, 2005.
- [31] T. F. Chan, G. H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.
- [32] T. F. Chan and J. J. Shen. *Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods*. SIAM, 2005.
- [33] E. Chouzenoux, A. Jezierska, J.-C. Pesquet, and H. Talbot. A convex approach for image restoration with exact Poisson–Gaussian likelihood. *SIAM Journal on Imaging Sciences*, 8(4):2662–2682, 2015.
- [34] R. Ciak, B. Shafei, and G. Steidl. Homogeneous penalizers and constraints in convex image restoration. *Journal of Mathematical Imaging and Vision*, 47(3):210–230, 2013.
- [35] P. G. Ciarlet. *Introduction to Numerical Linear Algebra and Optimisation*. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge, 1989.
- [36] P. L. Combettes and B. C. Vũ. Variable metric forward–backward splitting with applications to monotone inclusions in duality. *Optimization*, 63(9):1289–1318, 2014.

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

- [37] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.
- [38] L. Condat. A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *Journal of Optimization Theory and Applications*, 158(2):460–479, 2013.
- [39] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007.
- [40] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. BM3D image denoising with shape-adaptive principal component analysis. In *SPARS'09-Signal Processing with Adaptive Sparse Structured Representations*, 2009.
- [41] J. Darbon and M. Sigelle. A fast and exact algorithm for total variation minimization. In *Pattern Recognition and Image Analysis*, pages 351–359. Springer, 2005.
- [42] J. Darbon and M. Sigelle. Image restoration with discrete constrained total variation. I. Fast and exact optimization. *Journal of Mathematical Imaging and Vision*, 26(3):261–276, 2006.
- [43] I. Daubechies, G. Teschke, and L. Vese. Iteratively solving linear inverse problems under general convex constraints. *Inverse Problems and Imaging*, 1(1):29–46, 2007.
- [44] J. C. De los Reyes and C.-B. Schönlieb. Image denoising: Learning the noise model via nonsmooth PDE-constrained optimization. *Inverse Problems and Imaging*, 7(4), 2013.
- [45] J. Delon and A. Desolneux. A patch-based approach for removing impulse or mixed Gaussian-impulse noise. *SIAM Journal on Imaging Sciences*, 6(2):1140–1174, 2013.
- [46] D. C. Dobson and C. R. Vogel. Convergence of an iterative method for total variation denoising. *SIAM Journal on Numerical Analysis*, 34(5):1779–1791, 1997.
- [47] B. Dong, H. Ji, J. Li, Z. Shen, and Y. Xu. Wavelet frame based blind image inpainting. *Applied and Computational Harmonic Analysis*, 32(2):268–279, 2012.
- [48] Y. Dong, M. Hintermüller, and M. M. Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *Journal on Mathematical Imaging and Vision*, 40(1):82–104, 2011.
- [49] I. Ekeland and R. Témam. *Convex Analysis and Variational Problems*, volume 28 of *Classics in Applied Mathematics*. SIAM, Philadelphia, PA, english edition, 1999.
- [50] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian. Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008.
- [51] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40, 1976.
- [52] R. Garnett, T. Huegerich, C. Chui, and W. He. A universal noise removal algorithm with an impulse detector. *IEEE Transactions on Image Processing*, 14(11):1747–1754, 2005.
- [53] O. Ghita and P. F. Whelan. A new GVF-based image enhancement formulation for use in the presence of mixed noise. *Pattern Recognition*, 43(8):2646–2658, 2010.
- [54] E. Gil-Rodrigo, J. Portilla, D. Miraut, and R. Suarez-Mesa. Efficient joint Poisson-Gauss restoration using multi-frame l2-relaxed-l0 analysis-based sparsity. In *18th IEEE International Conference on Image Processing (ICIP 2011)*, pages 1385–1388. IEEE, 2011.
- [55] E. Giusti. *Minimal Surfaces and Functions of Bounded Variation*, volume 80 of *Monographs in Mathematics*. Birkhäuser Verlag, Basel, 1984.

## II. Parameter Selection Methods for Total Variation Models

- [56] R. Glowinski. *Numerical Methods for Nonlinear Variational Problems*. Springer-Verlag, New York, 1984.
- [57] T. Goldstein and S. Osher. The split Bregman method for  $L_1$ -regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.
- [58] Z. Gong, Z. Shen, and K.-C. Toh. Image restoration with mixed or unknown noises. *Multiscale Modeling & Simulation*, 12(2):458–487, 2014.
- [59] M. Hintermüller and K. Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM Journal on Applied Mathematics*, 64(4):1311–1333, 2004.
- [60] M. Hintermüller and A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1/L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.
- [61] M. Hintermüller and M. M. Rincon-Camacho. Expected absolute value estimators for a spatially adapted regularization parameter choice rule in  $L^1$ -TV-based image restoration. *Inverse Problems*, 26(8):085005, 30, 2010.
- [62] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.
- [63] Y.-M. Huang, M. K. Ng, and Y.-W. Wen. Fast image restoration methods for impulse and Gaussian noises removal. *IEEE Signal Processing Letters*, 16(6):457–460, 2009.
- [64] B. G. Jeong, B. C. Kim, Y. H. Moon, and I. K. Eom. Simplified noise model parameter estimation for signal-dependent noise. *Signal Processing*, 96:266–273, 2014.
- [65] A. Jezierska, J.-C. Pesquet, H. Talbot, and C. Chaux. Iterative Poisson-Gaussian noise parametric estimation for blind image denoising. In *IEEE International Conference on Image Processing (ICIP 2014)*, pages 2819–2823. IEEE, 2014.
- [66] N. Komodakis and J.-C. Pesquet. Playing with duality: An overview of recent primal-dual approaches for solving large-scale optimization problems. *IEEE Signal Processing Magazine*, 32(6):31–54, 2015.
- [67] K. Kunisch and T. Pock. A bilevel optimization approach for parameter learning in variational models. *SIAM Journal on Imaging Sciences*, 6(2):938–983, 2013.
- [68] A. Langer. Automated parameter selection for total variation minimization in image restoration. *arXiv preprint arXiv:1509.07442*, 2015.
- [69] M. Lebrun, A. Buades, and J.-M. Morel. A nonlocal bayesian image denoising algorithm. *SIAM Journal on Imaging Sciences*, 6(3):1665–1688, 2013.
- [70] B. Li, Q. Liu, J. Xu, and X. Luo. A new method for removing mixed noises. *Science China Information Sciences*, 54(1):51–59, 2011.
- [71] Y.-R. Li, L. Shen, D.-Q. Dai, and B. W. Suter. Framelet algorithms for de-blurring images corrupted by impulse plus Gaussian noise. *IEEE Transactions on Image Processing*, 20(7):1822–1837, 2011.
- [72] J. Liu, X.-C. Tai, H. Huang, and Z. Huan. A weighted dictionary learning model for denoising images corrupted by mixed noise. *IEEE Transactions on Image Processing*, 22(3):1108–1120, 2013.
- [73] R. W. Liu, L. Shi, S. C. H. Yu, and D. Wang. Box-constrained second-order total generalized variation minimization with a combined  $L^{1,2}$  data-fidelity term for image reconstruction. *Journal of Electronic Imaging*, 24(3):033026–033026, 2015.

## 10. Automated Parameter Selection in the $L^1$ - $L^2$ -TV Model

- [74] E. López-Rubio. Restoration of images corrupted by Gaussian and uniform impulsive noise. *Pattern Recognition*, 43(5):1835–1846, 2010.
- [75] Y. Meyer. *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations: the Fifteenth Dean Jacqueline B. Lewis Memorial Lectures*, volume 22. American Mathematical Soc., 2001.
- [76] T. B. Moeslund. *Introduction to Video and Image Processing: Building Real Systems and Applications*. Springer Science & Business Media, 2012.
- [77] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1, Series. A):127–152, 2005.
- [78] M. K. Ng, P. Weiss, and X. Yuan. Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods. *SIAM Journal on Scientific Computing*, 32(5):2710–2736, 2010.
- [79] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40(3):965–994, 2002.
- [80] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1-2):99–120, 2004.
- [81] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.
- [82] S. Peng and L. Lucke. Fuzzy filtering for mixed noise removal during image processing. In *Proceedings of the Third IEEE Conference on Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence*, pages 89–93. IEEE, 1994.
- [83] A. L. Peressini, F. E. Sullivan, and J. J. Uhl. *The Mathematics of Nonlinear Programming*, volume 1. Springer New York, 1988.
- [84] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.
- [85] Y. Shen, B. Han, and E. Braverman. Removal of mixed Gaussian and impulse noise using directional tensor product complex tight framelets. *Journal of Mathematical Imaging and Vision*, pages 1–14, 2015.
- [86] T. Teuber, G. Steidl, and R. H. Chan. Minimization and parameter estimation for seminorm regularization models with  $\mathcal{L}$ -divergence constraints. *Inverse Problems*, 29(3):035007, 2013.
- [87] L. Vese. A study in the BV space of a denoising-deblurring variational problem. *Applied Mathematics & Optimization*, 44(2):131–161, 2001.
- [88] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
- [89] P. Weiss, L. Blanc-Féraud, and G. Aubert. Efficient schemes for total variation minimization under constraints in image processing. *SIAM Journal on Scientific Computing*, 31(3):2047–2080, 2009.
- [90] Y.-W. Wen and R. H. Chan. Parameter selection for total-variation-based image restoration using discrepancy principle. *IEEE Transactions on Image Processing*, 21(4):1770–1781, 2012.
- [91] Y. Xiao, T. Zeng, J. Yu, and M. K. Ng. Restoration of images corrupted by mixed Gaussian-impulse noise via  $\mathcal{L}_1$  -  $\mathcal{L}_0$  minimization. *Pattern Recognition*, 44(8):1708–1720, 2011.
- [92] M. Yan. Restoration of images corrupted by impulse noise and mixed Gaussian impulse noise using blind inpainting. *SIAM Journal on Imaging Sciences*, 6(3):1227–1245, 2013.

## **II. Parameter Selection Methods for Total Variation Models**

- [93] J. X. Yang and H. R. Wu. Mixed Gaussian and uniform impulse noise analysis using robust estimation for digital images. In *16th International Conference on Digital Signal Processing, 2009*, pages 1–5. IEEE, 2009.
- [94] M. Zhu and T. Chan. An efficient primal-dual hybrid gradient algorithm for total variation image restoration. *UCLA CAM Report*, pages 08–34, 2008.

# 11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise

**Bibliographic Note:** The content of this chapter is published in [AL10]: A. Langer. Locally adaptive total variation for removing mixed Gaussian-impulse noise. *International Journal of Computer Mathematics*, 2018.

**Summary:** The minimization of a functional consisting of a combined  $L^1/L^2$  data fidelity term and a total variation regularization term with a locally varying regularization parameter for the removal of mixed Gaussian-impulse noise is considered. Based on a related locally constrained optimization problem, algorithms for automatically selecting the spatially varying parameter are presented. Numerical experiments for image denoising are shown, which demonstrate that the locally varying parameter selection algorithms are able to generate solutions which are of higher restoration quality than solutions obtained with scalar parameters.

## 11.1. Introduction

Due to several reasons observed images are often contaminated by different types of noise and may be additionally distorted by some measurement device. In this paper we consider images which are containing a mixture of Gaussian and impulse noise, where we assume that first Gaussian noise and then impulse noise is added to the image. This might be, for example, the case if in the process of image acquisition Gaussian noise occurs and later while image transmission impulse noise is added. In this setting an observation  $g$  might be modeled as

$$\begin{cases} \bar{g} = T\hat{u} + \eta \\ g = \mathcal{N}(\bar{g}), \end{cases}$$

where  $\hat{u}$  is the unknown original image,  $T$  is a linear bounded operator modeling the image formation device,  $\eta$  is oscillatory with zero mean and standard deviation  $\sigma$  representing Gaussian white noise, and  $\mathcal{N}$  represents impulse noise. In most of the applications impulse noise is either modeled as salt-and-pepper or random-valued impulse noise. Assume that  $\bar{g}$  lies in the dynamic range  $[d_{\min}, d_{\max}]$ , i.e.,  $d_{\min} \leq \bar{g} \leq d_{\max}$  almost everywhere. Then in the case of salt-and-pepper noise

$$g(x) = \begin{cases} d_{\min} & \text{with probability } s \\ d_{\max} & \text{with probability } s \\ \bar{g}(x) & \text{with probability } 1 - 2s \end{cases}$$

where  $s \in [0, 1/2)$ , and in the case of random valued impulse noise

$$g(x) = \begin{cases} \rho & \text{with probability } r \in [0, 1) \\ \bar{g}(x) & \text{with probability } 1 - r, \end{cases}$$

## II. Parameter Selection Methods for Total Variation Models

where  $\rho$  is a uniformly distributed random variable in the image intensity range  $[d_{\min}, d_{\max}]$ ; see for example [7]. Note, that if first impulse noise and then Gaussian noise is added, a different observation is formed; see [28].

There exists a variety of different approaches for removing mixed Gaussian-impulse noise which usually start by estimating or detecting outliers (impulse noise) in the image and then adapt or use a Gaussian noise removal strategy; see for example [7, 16, 26, 32, 37, 41, 43]. In general, algorithms for Gaussian plus impulse noise removal may be classified in the following way: filter approaches [15, 36, 43], regularization based approaches [7, 13, 16, 26, 32, 38, 41, 42], Bayesian-based approaches [35], and patch-based approaches [12, 29, 33]. In this paper we consider a pure regularization approach by minimizing a functional consisting of a combined  $L^1$ - $L^2$ -data-fidelity term and a total variation regularization term, introduced in [20]. It has been demonstrated that this optimization problem is suited to the task of removing mixed Gaussian-impulse noise; see [20, 28]. In this approach, which we call  $L^1$ - $L^2$ -TV model, an image is restored by solving

$$\min_{u \in BV(\Omega)} \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \lambda \int_{\Omega} |Du|, \quad (11.1)$$

where  $\Omega \subset \mathbb{R}^2$  is an open bounded domain with Lipschitz boundary,  $T_i : L^2(\Omega) \rightarrow L^2(\Omega)$  is a bounded linear operator,  $g_i \in L^2(\Omega)$  is a given datum,  $\alpha_i \geq 0$  for  $i = 1, 2$  with  $\alpha_1 + \alpha_2 > 0$ , and  $\lambda > 0$ . Here,  $\int_{\Omega} |Du|$  denotes the total variation (TV) of  $u$  in  $\Omega$ ; see [3, 18] for more details. We note, that one parameter in (11.1) can be actually omitted. However, for good reasons, which will be clear later, we would like to keep the  $L^1$ - $L^2$ -TV model in this form.

For removing a mixture of Gaussian and impulse noise from a single image  $g$  one typically sets  $T_1 = T_2$  and  $g_1 = g_2 = g$  in (11.1). In this setting it is easy to see that the  $L^1$ - $L^2$ -TV model (11.1) is a generalization of the  $L^1$ -TV model (if  $\alpha_2 = 0$ ) and the  $L^2$ -TV model (if  $\alpha_1 = 0$ ). Modifications of the  $L^1$ - $L^2$ -TV model have been presented in [19], where the total variation is replaced by  $\|Wu\|_1$  with  $W$  being a wavelet tight frame transform, and in [34], where the second order total generalized variation [6] has been used as regularization term and box-constraints, which assure that the reconstruction lies in the respective dynamic range, are incorporated. It is worth to mention, that very recently in [8] a different regularization approach using an infimal convolution data fidelity term is proposed for mixed noise removal.

The minimizer of the  $L^1$ - $L^2$ -TV model highly depends on the proper choice of  $\alpha_i$ ,  $i = 1, 2$  and  $\lambda$ . In particular, if  $\alpha_1$  and  $\alpha_2$  are small compared to  $\lambda$ , then an over-smoothed reconstruction is obtained, which not only removes noise but also eliminates details in the image. On the contrary, if  $\alpha_1$  and  $\alpha_2$  are large in comparison to  $\lambda$ , then the obtained solution fits the given data properly but retains noise. Note, that  $\alpha_1$  and  $\alpha_2$  additionally weight the importance of the  $L^1$ -term and  $L^2$ -term. In particular, we expect  $\alpha_1$  to be large if the noise in the image is impulse noise dominated, while for Gaussian noise dominated images  $\alpha_2$  should be sufficiently large. Hence for fixed  $\lambda$  a good reconstruction can be achieved by choosing  $\alpha_1$  and  $\alpha_2$  such that a good compromise of the aforementioned effects are made. For  $\lambda = 1$  in [34] it is suggested to select the parameters according to the variance  $\sigma^2$  of the Gaussian noise and the energy of the impulse noise, i.e.,

$$\alpha_1 = \frac{E_I}{E_I + \sigma^2} \quad \text{and} \quad \alpha_2 = \frac{\sigma^2}{E_I + \sigma^2}, \quad (11.2)$$

where  $E_I = s$  for salt-and-pepper noise and  $E_I = \frac{r}{3}$  for random-valued impulse noise. This parameter choice rule is used in [28] as an initial guess for an automated parameter selection

## 11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise

algorithm, which is demonstrated to optimize the parameters further with respect to image quality measures. We emphasize, that the parameter selection algorithm of [28] chooses only scalar parameters.

Note, that images usually consist of large uniform regions as well as parts with fine details. Hence, scalar regularization parameters might not be the best choice; see [14, 27]. In particular, it is shown for the  $L^1$ -TV and  $L^2$ -TV model, that with a spatially varying weight better reconstructions (in the sense of certain quality measures) than with a globally constant parameter can be obtained, see for example [14, 25, 27]. We expect a similar behavior of the  $L^1$ - $L^2$ -TV model. In this vein, in order to enhance image details while removing noise considerably in uniform regions we consider the minimization problem

$$\min_{u \in BV(\Omega)} \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}^2 + \int_{\Omega} \lambda(x) |Du|, \quad (11.3)$$

where  $\lambda : \Omega \rightarrow \mathbb{R}^+$  is a function. In this paper we address the question, how to choose  $\lambda$  automatically in a suitable way.

Let us mention, that spatially varying choices of parameters for the  $L^2$ -TV and  $L^1$ -TV model are for example introduced in [2, 4, 9, 14, 17, 21, 23, 24, 25, 27, 31, 39]. Other data terms with locally varying parameters have for example been considered in [11, 30]. However, the combined  $L^1$ - $L^2$ -TV model has not yet been considered with respect to spatially varying parameter choices. To the best of our knowledge, in this paper we present the first approach in this direction. In order to do so we utilize the locally adaptive parameter approach of [27], which is based on the discrepancy principle and developed solely for the  $L^2$ -TV and  $L^1$ -TV model, and adapt it to our model. In this vein we relate (11.3) to a locally constrained optimization problem, see (11.4) below. We consider two different approaches on how to choose the local bounds in the constituted constraints and compare them numerically by experiments.

The contribution of the present paper is two-sided: Firstly, we present the first attempt of choosing a spatially adaptive parameter in the  $L^1$ - $L^2$ -TV model. Secondly, we compare two different approaches on how to choose the local bounds. In particular, this allows us to show the full strength of a locally varying parameter and shows at the same time the importance of properly chosen local bounds.

### 11.1.1. Locally Constrained Minimization Problem

In order to obtain a suitable  $\lambda$  in (11.3), we formulate, as in [14, 25, 27], a locally constrained optimization problem in the following way:

$$\min_{u \in BV(\Omega)} \int_{\Omega} |Du| \quad \text{subject to (s.t.)} \quad \mathcal{S}(u) \leq \mathcal{B} \quad \text{almost everywhere (a.e.) in } \Omega, \quad (11.4)$$

where  $\mathcal{S}(u)(\cdot) := \int_{\Omega} w(\cdot, y) \mathcal{H}(u)(y) dy$  with  $w$  being a normalized localization filter, i.e.,  $w \in L^{\infty}(\Omega \times \Omega)$ , and  $w \geq 0$  on  $\Omega \times \Omega$  with

$$\begin{aligned} & \int_{\Omega} \int_{\Omega} w(x, y) dy dx = 1, \\ & \int_{\Omega} \int_{\Omega} w(x, y) |\phi|(y) dy dx \geq \epsilon \|\phi\|_{L^1(\Omega)} \quad \text{for all } \phi \in L^1(\Omega) \quad \text{and} \\ & \int_{\Omega} \int_{\Omega} w(x, y) \phi^2(y) dy dx \geq \epsilon \|\phi\|_{L^2(\Omega)}^2 \quad \text{for all } \phi \in L^2(\Omega) \end{aligned} \quad (11.5)$$

## II. Parameter Selection Methods for Total Variation Models

for some  $\epsilon > 0$  independent of  $\phi$ , cf. [25], and  $\mathcal{H}(u)(y) := \alpha_1|T_1u - g_1|(y) + \alpha_2|T_2u - g_2|^2(y)$ . Here,  $\mathcal{B} \in \mathbb{R}$  is some value depending on the type of noise. Although the value  $\mathcal{B}$  is here a fixed constant, in our numerical experiments we report on results where  $\mathcal{B}$  is chosen empirically based on some approximation of the true image. Moreover, we define the feasible set

$$U := \{u \in BV(\Omega) : \mathcal{S}(u) \leq \mathcal{B} \text{ a.e. in } \Omega\}.$$

A function  $w$  which satisfies the conditions in (11.5) is, for instance, the mean filter defined as

$$w(x, y) := \begin{cases} \frac{1}{\omega_\delta} & \text{if } |y - x|_{\ell^\infty} \leq \frac{\omega}{2}, \\ \delta & \text{otherwise,} \end{cases}$$

with  $x \in \Omega$  being fixed,  $\omega > 0$  sufficiently small (representing the essential width of the filter window),  $0 < \delta \ll 1$  and  $\omega_\delta$  chosen such that  $\int_{\Omega} \int_{\Omega} w(x, y) dy dx = 1$ ; see [14, 25].

Next, we are going to show the existence of minimizers of (11.4) by using the following lemma.

**Lemma 11.1.** *Let  $\mathcal{Q}(u) = \mathcal{R}(u) + \int_{\Omega} \mathcal{S}(u)(x) dx$ . Assume there exist  $i \in \{1, 2\}$  such that  $T_i$  does not annihilate constant functions, i.e.,  $T_i \chi_{\Omega} \neq 0$ , where  $\chi_{\Omega}(x) = 1$  if  $x \in \Omega$ . Then  $\|u\|_{BV} \rightarrow \infty$  implies  $\mathcal{Q}(u) \rightarrow \infty$ .*

By noting that the conditions in (11.5) hold, the proof of this lemma follows the lines of the proof of [28, Lemma 3.1].

**Theorem 11.2.** *Assume there exist  $i \in \{1, 2\}$  such that  $T_i$  does not annihilate constant functions and  $U \neq \emptyset$ . Then the problem in (11.4) has a solution  $u \in BV(\Omega)$ .*

Using Lemma 11.1 the proof of this statement is analog to the proof of [14, Theorem 2] and [25, Theorem 2].

### 11.2. Locally Adaptive Algorithm

In what follows we consider a discrete image domain again denoted by  $\Omega$  containing  $N_1 \times N_2$  pixels,  $N_1, N_2 \in \mathbb{N}$ . The considered function spaces are  $X \in \mathbb{R}^{N_1 \times N_2}$  and  $Y = X \times X$ . For  $u \in X$  we use the following norms

$$\|u\|_{\ell^\tau(\Omega)}^\tau = \sum_{x \in \Omega} |u(x)|^\tau$$

for  $\tau \in \{1, 2\}$ . Then the discrete version of the functional in (11.3) writes as

$$J(u, \lambda) := \alpha_1 \|T_1 u - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{\ell^2(\Omega)}^2 + \sum_{x \in \Omega} \lambda(x) |\nabla u(x)|_{l^2} \quad (11.6)$$

where  $|y|_{l^2} = \sqrt{y_1^2 + y_2^2}$  for every  $y = (y_1, y_2)$ ,  $\lambda : \Omega \rightarrow \mathbb{R}^+$  is a discrete function,  $T_i : X \rightarrow X$  is a bounded linear operator,  $g_i \in X$  is a given datum, and  $\alpha_i \geq 0$  for  $i = 1, 2$  with  $\alpha_1 + \alpha_2 > 0$ . Here and in the sequel  $\nabla$  represents the discrete gradient (discretized by using forward differences; cf. [11, 22]) and  $\operatorname{div} = -\nabla^T$ . Moreover, we assume that  $T_i$  is diagonal for  $i = 1, 2$ , i.e., there exists  $\tilde{T}_i \in X$  associated with  $T_i$  such that  $T_i u = \tilde{T}_i \circ u$  (Hadamard product) with  $[\tilde{T}_i \circ u](x) = \tilde{T}_i(x)u(x)$  for any  $u \in X$  and  $x \in \Omega$ . This is for example the case when  $T_i = I$  or  $T_i = \chi_D$ , where  $\chi_D$  denotes the characteristic function of the domain  $D \subset \Omega$ , which are the two relevant cases for our numerical experiments.

### 11.2.1. The Locally Constrained Problem

For a reasonable choice of  $\lambda$  in (11.6) we consider the following optimization problem with local constraints, cf. (11.4):

$$\min_{u \in X} \|\nabla u\|_{\ell^1(\Omega)} \quad \text{s.t.} \quad S(u)(x) \leq B \quad \text{for all } x \in \Omega, \quad (11.7)$$

where

$$S(u)(x) := \frac{1}{|\mathcal{I}(x)|} \sum_{y \in \mathcal{I}(x)} \sum_{\tau=1}^2 \alpha_\tau |(T_\tau u)(y) - g_\tau(y)|^\tau$$

denotes a kind of local residual at  $x \in \Omega$  with  $\mathcal{I}(x)$  being some suitable set of pixels around  $x$  of size  $|\mathcal{I}(x)|$  (i.e.  $|\mathcal{I}(x)|$  denotes the number of pixels in  $\mathcal{I}(x)$ ) and  $B \geq 0$  denotes the noise-level of the respective noise, which may depend on the true (unknown) image  $\hat{u}$ ; cf. (11.4). For more details on the choice of  $B$  we refer to Section 11.2.2 below.

In the sequel, as in [27], we set

$$\mathcal{I}(x) := \left\{ y \in \Omega : |x - y| \leq \frac{\omega - 1}{2} \right\},$$

that is,  $\mathcal{I}(x) \subset \Omega$  for all  $x \in \Omega$ , and  $\omega$  being an odd number determining the size. With this separation of the image domain we obtain the following results.

**Proposition 11.3.** *If  $u$  is a solution of (11.7), then we have that*

$$H(u) := \alpha_1 \|T_1 u - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{\ell^2(\Omega)}^2 \leq B|\Omega|.$$

*Proof.* Let  $u$  be a minimizer of (11.7), then we have that

$$\begin{aligned} B|\Omega| &= |\Omega|(\alpha_1 \nu_1 + \alpha_2 \nu_2) \geq \sum_{x \in \Omega} S(u)(x) = \sum_{x \in \Omega} \frac{1}{|\mathcal{I}(x)|} \sum_{y \in \mathcal{I}(x)} \sum_{\tau=1}^2 \alpha_\tau |(T_\tau u)(y) - g_\tau(y)|^\tau \\ &= \sum_{x \in \Omega} \sum_{\tau=1}^2 \alpha_\tau |(T_\tau u)(x) - g_\tau(x)|^\tau = \alpha_1 \|T_1 u - g_1\|_{\ell^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{\ell^2(\Omega)}^2 \end{aligned}$$

where we used that, due to the sum over  $x \in \Omega$  each picture element  $y \in \mathcal{I}(x)$  occurs exactly  $|\mathcal{I}(x)|$ -times.  $\square$

From this proposition it follows that a minimizer of the locally constrained problem (11.7) also satisfies the constraint of the globally constrained problem

$$\min_{u \in X} \|\nabla u\|_{\ell^1(\Omega)} \quad \text{s.t.} \quad H(u) \leq B|\Omega| \quad (11.8)$$

but is in general of course not a solution of (11.8). Consequently, [27, Proposition 4.2] holds also in our framework, which we recall next for completeness.

**Proposition 11.4.** *Let  $u_s$  and  $u_l$  be solutions of (11.8) and (11.7) respectively. Then we have that*

$$\|\nabla u_s\|_{\ell^1(\Omega)} \leq \|\nabla u_l\|_{\ell^1(\Omega)}.$$

This statement says that the total variation of the solution of (11.7) cannot be smaller than the total variation of the minimizer of (11.8). Meaning that the locally constrained problem is expected to preserve more details than the globally constrained problem. However, we keep in mind that noise may be interpreted as fine details.

## II. Parameter Selection Methods for Total Variation Models

### 11.2.2. On the Choice of the Local Bounds

In this paper we consider two different approaches of choosing the local bounds  $B$ , which we are going to compare numerically in Section 11.3.

#### 11.2.2.1. Approach 1

Assume that for an image corrupted by mixed Gaussian-impulse noise the probability  $s$  or  $r$ , depending which type of impulse noise (i.e., salt-and-pepper or random-valued impulse noise) is present in the image, and the standard deviation  $\sigma$  of Gaussian noise is at hand. Then the expected absolute value (EAV)  $\nu_1$  and the variance  $\nu_2$  of the mixed Gaussian-impulse noise can be estimated as in [28]. More precisely, knowing  $s, r$  and  $\sigma$  allows us to compute the EAV of each single present noise type. Then  $\nu_1$  is the sum of the EAV of the impulse noise and Gaussian noise and  $\nu_2$  is the sum of the variances of both noises. Then the local bound  $B$  is set to  $B = \alpha_1\nu_1 + \alpha_2\nu_2$ . The motivation of the above choice of  $B$  is, that we wish for a restoration  $u$  such that

$$\|T_1u - g_1\|_{\ell^1(\mathcal{I}(x))} \leq \nu_1|\mathcal{I}(x)| \quad \text{and} \quad \|T_2u - g_2\|_{\ell^2(\mathcal{I}(x))}^2 \leq \nu_2|\mathcal{I}(x)| \quad (11.9)$$

and hence  $\|T_1u - g_1\|_{\ell^1(\mathcal{I}(x))} + \|T_2u - g_2\|_{\ell^2(\mathcal{I}(x))}^2 \leq (\nu_1 + \nu_2)|\mathcal{I}(x)|$ . We are aware, that a solution of (11.7) in general does not fulfill the inequalities in (11.9). However, we expect for a minimizer of (11.7), that these inequalities are at least not too much violated. This expectation is based on the fact, that if a solution of (11.7) violates one of the inequalities in (11.9) the other inequality holds automatically. Further, if  $T_1 = T_2$  and  $g_1 = g_2$ , which is the typical setting for restoring a single image, then this fact is additionally accompanied by the circumstance, that the  $\ell^1$ -norm and the squared  $\ell^2$ -norm behave similarly (they are both convex and share the same minimum). That is, if one of these norms is increasing the other does as well.

As one sees from the calculation in [28],  $\nu_1$  and  $\nu_2$  may depend on the original image  $\hat{u}$ . Whenever this happens for at least one  $\nu_\tau$ ,  $\tau \in \{1, 2\}$ , problem (11.7) results in a quite nonlinear problem, since  $B$  becomes  $B(u)$ . Instead of considering nonlinear constraints we choose a reference image  $\tilde{u}$  and compute an approximation  $B = B(\tilde{u})$ . Since our proposed algorithms are of iterative nature (see LATV- and pLATV-algorithm below), the reference image  $\tilde{u}$  is chosen as the current approximation (see LATV- and pLATV-algorithm below). Then, with this new value  $B$  we are solving (11.7).

#### 11.2.2.2. Approach 2

We suppose that we know in each window  $\mathcal{I}(x)$ ,  $x \in \Omega$ , the true expected absolute value and standard deviation of the mixed noise. Then (11.7) writes as

$$\min_{u \in X} \|\nabla u\|_{\ell^1(\Omega)} \quad \text{s.t.} \quad S(u)(x) \leq B(x) \quad \text{for all } x \in \Omega, \quad (11.10)$$

where  $B(x) = c(\alpha_1\nu_1(x) + \alpha_2\nu_2(x))$  with  $\nu_1(x)$  and  $\nu_2(x)$  being the known EAV and standard deviation in  $\mathcal{I}(x)$  for  $x \in \Omega$  and  $c > 0$  is a constant experimentally obtained. In particular in our numerics it turns out that  $c = 0.95$  does a good job for all considered examples and hence we set it exactly to that value for all our experiments.

## 11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise

Similar to Proposition 11.3 we can show that a minimizer of (11.10) satisfies the condition of the globally constrained problem

$$\min_{u \in X} \|\nabla u\|_{\ell^1(\Omega)} \quad \text{s.t.} \quad H(u) \leq \sum_{x \in \Omega} B(x).$$

Note, that  $\sum_{x \in \Omega} B(x) = B|\Omega|$  if  $B(x) =: B$  is constant for all  $x \in \Omega$ .

We summarize that the main difference between approach 1 and approach 2 is, that in approach 1 the bound  $B$  is constant in whole  $\Omega$ , while for approach 2 it is spatially varying.

Our numerical experiments demonstrate that approach 2 is superior to approach 1. However, obtaining the exact values  $\nu_1(x)$  and  $\nu_2(x)$  for any  $x \in \Omega$  seems very challenging and is not part of the present paper, but will be considered in future research.

In the sequel, in order to keep the paper concise, we will write  $B(x)$  or  $B(\tilde{u})(x)$  independently whether the bound is constant for any  $x \in \Omega$  or not.

### 11.2.3. Local Parameter Selection

Note, that for small  $\alpha_1, \alpha_2 > 0$  (small compared to  $\lambda$ ) the minimization of (11.1) yields an over-smoothed restoration  $u_{\alpha_1, \alpha_2}$  and we expect  $H(u_{\alpha_1, \alpha_2}) > \sum_{x \in \Omega} B$ . Similar, if  $S(u)(x) > B(x)$

we suppose that this is because  $u$  is over-smoothed in  $\mathcal{I}(x)$  and hence we intend to decrease  $\lambda$  in the local region  $\mathcal{I}(x)$ . By similar considerations as in [27] we obtain the following locally adapted algorithm:

**Algorithm 10** (LATV-Algorithm). *Initialize  $p := p_0 > 0$ ,  $\lambda_0 > 0$  and set  $n = 0$ ;*

1) *Compute  $u_{\lambda_n} \in \arg \min_{u \in X} J(u, \lambda_n)$*

2) (a) *If  $H(u_{\lambda_0}) > \sum_{x \in \Omega} B(u_{\lambda_0})(x)$ , then set*

$$f(x) := \max \{S(u_{\lambda_n})(x), B(u_{\lambda_n})(x)\}$$

(b) *If  $H(u_{\lambda_0}) < \sum_{x \in \Omega} B(u_{\lambda_0})(x)$ , then set*

$$f(x) := \max \{\min \{S(u_{\lambda_n})(x), B(u_{\lambda_n})(x)\}, \varepsilon\}$$

3) *Update*

$$\lambda_{n+1}(x) := \frac{\lambda_n(x)}{|\mathcal{I}(x)|} \sum_{y \in \mathcal{I}(x)} \left( \frac{B(u_{\lambda_n})(y)}{f(y)} \right)^p \quad \text{for all } x \in \Omega.$$

4) *Stop or set  $n := n + 1$  and return to step 1).*

In order to ensure that  $f(x) > 0$  for all  $x \in \Omega$  in every iteration, we introduce the small constant  $\epsilon > 0$ , e.g., we set  $\epsilon = 10^{-14}$  in our experiments.

If  $H(u_{\lambda_0}) > \sum_{x \in \Omega} B(u_{\lambda_0})(x)$ , we terminate the algorithm as soon as the residual  $H(u_{\lambda_n}) < \sum_{x \in \Omega} B(u_{\lambda_n})(x)$  for the first time and set the desired locally varying  $\lambda^* = \lambda_n$ . If  $H(u_{\lambda_0}) <$

## II. Parameter Selection Methods for Total Variation Models

$\sum_{x \in \Omega} B(u_{\lambda_0})(x)$ , we stop the algorithm as soon as the residual  $H(u_{\lambda_n}) > \sum_{x \in \Omega} B(u_{\lambda_n})(x)$  for the first time and set the desired locally varying  $\lambda^* = \lambda_{n-1}$ , since  $H(u_{\lambda_{n-1}}) \leq \sum_{x \in \Omega} B(u_{\lambda_{n-1}})(x)$ .

From [27] we have the following monotonicity property.

**Proposition 11.5** ([27, Proposition 4.4]). *The LATV-algorithm generates a sequence  $(\lambda_n)_n$  such that  $(\sum_{x \in \Omega} \lambda_n(x))_n$  is monotone.*

Similar as in [27] the power  $p$  might be chosen adaptively leading to the following algorithm:

**Algorithm 11** (pLATV-Algorithm). *Initialize  $p := p_0 > 0$ ,  $\lambda_0 > 0$  and set  $n = 0$ ;*

1) Compute  $u_{\lambda_0} \in \arg \min_{u \in X} J(u, \lambda_0)$

2) (a) If  $H(u_{\lambda_0}) > \sum_{x \in \Omega} B(u_{\lambda_0})(x)$ , then set

$$f(x) := \max \{S(u_{\lambda_n})(x), B(u_{\lambda_n})(x)\}.$$

(b) If  $H(u_{\lambda_0}) < \sum_{x \in \Omega} B(u_{\lambda_0})(x)$ , then set

$$f(x) := \max \{\min \{S(u_{\lambda_n})(x), B(u_{\lambda_n})(x)\}, \varepsilon\}.$$

3) Update

$$\lambda_{n+1}(x) := \frac{\lambda_n(x)}{|\mathcal{I}(x)|} \sum_{y \in \mathcal{I}(x)} \left( \frac{B(u_{\lambda_n})(y)}{f(y)} \right)^p \quad \text{for all } x \in \Omega.$$

4) Compute  $u_{\lambda_{n+1}} \in \arg \min_{u \in X} J(u, \lambda_{n+1})$

5) (a) if  $H(u_{\lambda_0}) > \sum_{x \in \Omega} B(u_{\lambda_0})(x)$

(i) if  $H(u_{\lambda_{n+1}}) \geq \sum_{x \in \Omega} B(u_{\lambda_{n+1}})(x)$ , go to step 6)

(ii) if  $H(u_{\lambda_{n+1}}) < \sum_{x \in \Omega} B(u_{\lambda_{n+1}})(x)$ , decrease  $p$ , e.g., set  $p = p/10$ , and go to step 2)

(b) if  $H(u_{\lambda_0}) \leq \sum_{x \in \Omega} B(u_{\lambda_0})(x)$

(i) if  $H(u_{\lambda_{n+1}}) \leq \sum_{x \in \Omega} B(u_{\lambda_{n+1}})(x)$ , go to step 6)

(ii) if  $H(u_{\lambda_{n+1}}) > \sum_{x \in \Omega} B(u_{\lambda_{n+1}})(x)$ , decrease  $p$ , e.g., set  $p = p/10$ , and go to step 2)

6) Stop or set  $n := n + 1$  and return to step 2).

This algorithm is terminated as soon as  $|H(u_{\lambda_n}) - \sum_{x \in \Omega} B(u_{\lambda_n})(x)| \leq 10^{-6}$  and  $H(u_{\lambda_n}) \leq \sum_{x \in \Omega} B(u_{\lambda_n})(x)$  for the first time. Additionally we stop iterating when  $p$  is less than machine precision, since then anyway no progress is to expect. Due to the adaptive choice of  $p$  we obtain a monotonic behavior of the sequence  $(\lambda_n)_n$ ; see [27, Proposition 4.5].

Note, that the LATV- and pLATV-algorithm do not adjust the parameters  $\alpha_1$  and  $\alpha_2$ , which are weighting the importance of the  $L^1$ - and  $L^2$ -norm. In particular, as already mentioned

## 11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise

above, we would like to choose  $\alpha_1$  large if the noise in the image is impulse noise dominated, while for Gaussian noise dominated images  $\alpha_2$  should be sufficiently large. In order to use suitable  $\alpha_1$  and  $\alpha_2$  we set them according to (11.2).

The LATV- and pLATV-algorithm only allows that  $\lambda_n(x)$  for all  $x$  is either not increased or not decreased. That is, it is not possible that  $\lambda_n(x_1)$  is increased and  $\lambda_n(x_2)$  is decreased for arbitrary  $x_1, x_2 \in \Omega$ . Hence the initial value  $\lambda_0$  has to be chosen sufficiently large or sufficiently small. In our experiments  $\lambda_0 = 1$  owns this property.

### 11.2.4. Primal-Dual Algorithm for Locally Adaptive Total Variation

Since  $T_i$ ,  $i = 1, 2$ , is diagonal, a minimizer of problem (11.3) is computed by utilizing the primal-dual approach proposed by Chambolle and Pock [10], which is already used for approximating a solution of the (scalar)  $L^1$ - $L^2$ -TV problem (11.1) in [1] in a finite element setting. For  $T_1 = T_2 = I$  the steepest descent algorithm runs as follows:

**Algorithm 12** (Primal-dual algorithm). *Initialize  $\tau, \rho > 0$ ,  $\theta \in [0, 1]$ ,  $\mathbf{p}_0 \in Y$ ,  $u_0 = \bar{u}_0 \in X$  and set  $n = 0$ ;*

$$1) \text{ Update } \mathbf{p}_{k+1}(x) = \frac{\mathbf{p}_k(x) + \sigma \nabla \bar{u}(x)}{\max\{\frac{1}{\lambda(x)}|\mathbf{p}_k(x) + \sigma \nabla \bar{u}(x)|, 1\}}$$

2) *Update*

$$u_{k+1}(x) = \begin{cases} z(x) - \beta & \text{if } z(x) - \beta > g_1(x), \\ z(x) + \beta & \text{if } z(x) + \beta < g_1(x), \\ g_1(x) & \text{else ,} \end{cases}$$

where  $z = \frac{1}{1+2\tau\alpha_2}(u_k + \tau \operatorname{div} \mathbf{p}_{k+1} + 2\tau\alpha_2 g_2)$  and  $\beta = \frac{\tau\alpha_1}{1+2\tau\alpha_2}$ .

3) *Update*  $\bar{u}_{k+1} = u_{k+1} + \theta(u_{k+1} - u_k)$ .

4) *Stop or set  $k := k + 1$  and return to step 1).*

If  $\theta = 1$  and  $\tau\sigma\|\nabla\|^2 < 1$ , then the convergence of this iterative scheme is ensured; see [10].

For  $T_1 = \chi_{\Omega_1}$  and  $T_2 = \chi_{\Omega_2}$ , which is the case in the example considered in section 11.3.3, where  $\Omega_1, \Omega_2 \subseteq \Omega$ , then we still use Algorithm 12 by just replacing in step 2  $z$  and  $\beta$  with  $z(x) = \frac{1}{1+2\tau\alpha_2\chi_{\Omega_2}(x)}(u_k(x) + \tau \operatorname{div} \mathbf{p}_{k+1}(x) + 2\tau\alpha_2\chi_{\Omega_2}(x)g_2(x))$  and  $\beta(x) = \frac{\tau\chi_{\Omega_1}(x)\alpha_1}{1+2\tau\chi_{\Omega_2}(x)\alpha_2}$  for all  $x \in \Omega$ , respectively.

## 11.3. Numerical Results

In the following we present numerical experiments for studying the behavior of the proposed algorithms (i.e., LATV- and pLATV-algorithm) with respect to its image restoration capabilities. We consider the LATV- and pLATV-algorithm with both types of local bounds described in Section 11.2.2 to which we will refer as approach 1 and approach 2.

The performance of these methods is compared quantitatively by means of peak signal-to-noise ratio (PSNR) [5], which is widely used as an image quality assessment measure and the mean structural similarity measure (MSSIM) [40], which relates to perceived visual quality better than PSNR. In general, when comparing PSNR and MSSIM, large values indicate better reconstruction than smaller values.

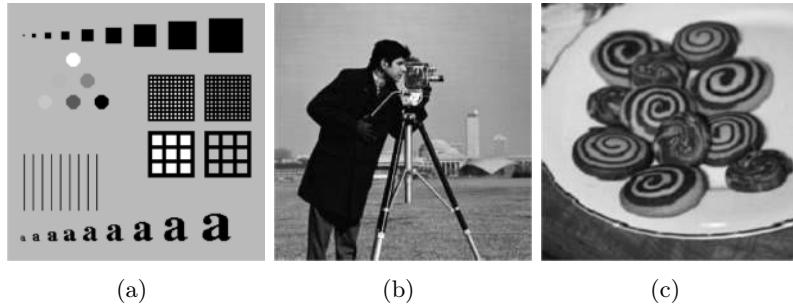
## II. Parameter Selection Methods for Total Variation Models

For our numerical studies we consider the images shown in Figure 11.1 of size  $256 \times 256$  pixels. The image intensity range of these images is scaled to  $[0, 1]$ .

In all our experiments the initial power  $p$  is chosen to be  $\frac{1}{2}$  and  $\lambda_0 = 1$ . Moreover, the parameters  $\alpha_1$  and  $\alpha_2$  are chosen according to (11.2). The optimization problem occurring in step 1 of the LATV-algorithm and in step 1 and step 4 of the pLATV-algorithm are solved via the primal-dual scheme in Algorithm 12. For the primal-dual algorithm we set  $\theta = 1$  and  $\sigma = \tau = (0.9/8)^{1/2}$ . Since we are using the discretization used in [10] with spacing-size 1,  $\|\nabla\|^2 = 8$  and hence  $\tau\sigma\|\nabla\|^2 < 1$ , which ensures the convergence of the used primal-dual algorithm. This algorithm is terminated as soon as the relative differences of two successive iterates are smaller than a certain tolerance, i.e., in our experiments as soon as

$$\frac{\|\mathbf{p}_k - \mathbf{p}_{k-1}\|_{\ell^1}}{\|\mathbf{p}_k\|_{\ell^1}} \leq 10^{-4} \quad \text{and} \quad \frac{\|u_k - u_{k-1}\|_{\ell^1}}{\|u_k\|_{\ell^1}} \leq 10^{-4}$$

for the first time.



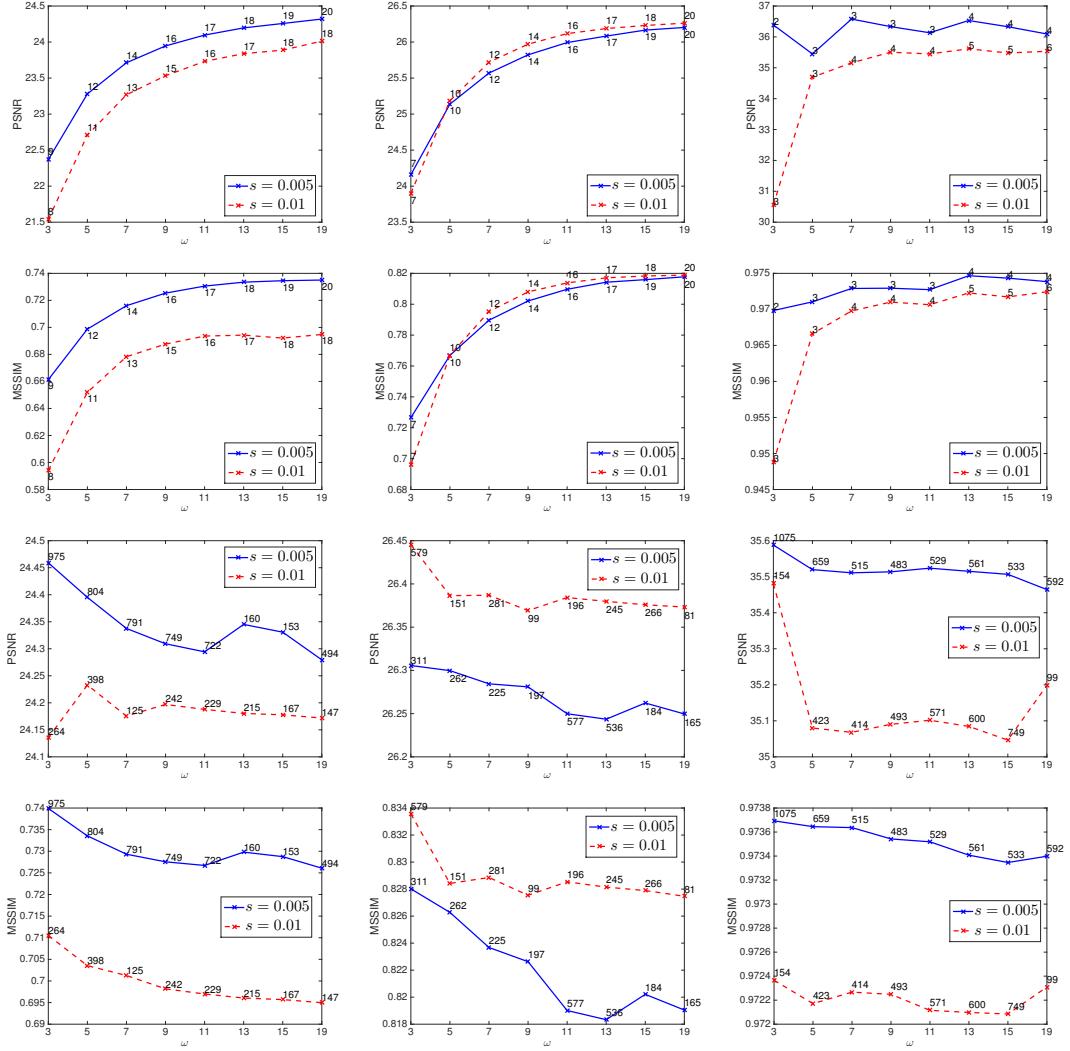
**Figure 11.1:** Original images: (a) phantom (b) cameraman (c) cookies

### 11.3.1. Dependency on the Window-Size

We test the influence of the window-size  $\omega$  on the restorations obtained by the proposed LATV- and pLATV-algorithm. To do so, we consider the cookies-image, see Figure 11.1(c), distorted by different levels of mixed Gaussian and salt-and-pepper noise and we vary  $\omega$  from 3 to 19. Figure 11.2 shows the PSNR and MSSIM of the respective restorations obtained via the LATV-algorithm and pLATV-algorithm using approach 1 while in Figure 11.3 we used approach 2. From these experiments we observe that larger window-sizes  $\omega$  usually decrease the PSNR and MSSIM of the reconstructions, while for the LATV-algorithm using approach 1 we observe exactly the opposite behavior. Similar observations are made for other images as well, e.g., the cameraman-image from Figure 11.1(b). This indicates to use a large window-size (e.g.  $\omega = 19$ ) in the LATV-algorithm with approach 1 and a very small window-size (e.g.,  $\omega = 3$ ) for all the other cases.

In Figure 11.2 and Figure 11.3 we also indicate the number of iterations needed till termination, which shows that the pLATV-algorithm needs a massive amount of more iterations than the LATV-algorithm. Moreover we observe that for the LATV-algorithm the number of iteration are increasing with increasing  $\omega$ , while this is in general not the case for the pLATV-algorithm, which anyway needs much more iterations till termination, due to the adaptivity of  $p$ .

## 11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise

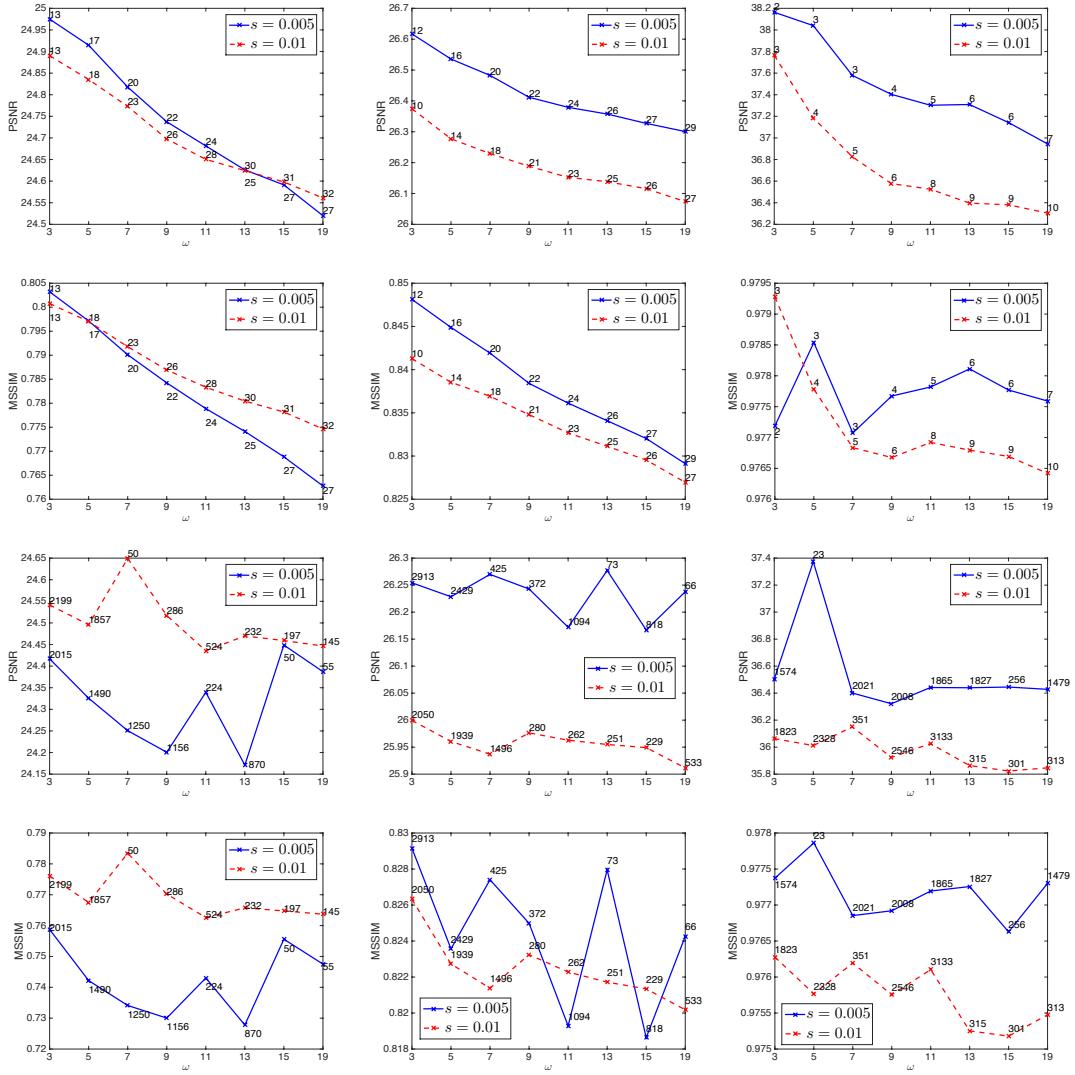


**Figure 11.2:** Restoration via the LATV-algorithm (first two rows) and via the pLATV-algorithm (last two rows) of the cookies-image corrupted with Gaussian white noise with  $\sigma = \sqrt{0.02}$  (left column),  $\sigma = 0.1$  (middle column),  $\sigma = 0.01$  (right column) and salt-and-pepper noise.

### 11.3.2. Comparison with Scalar Parameter

Now we compare the performance of the LATV- and pLATV-algorithm with two scalar parameter choice approaches. The first one is the algorithm proposed in [28], which automatically selects the scalar parameters  $\alpha_1$  and  $\alpha_2$  in (11.1) with  $\lambda = 1$ . In the sequel we refer to this algorithm as the  $\alpha_1$ - $\alpha_2$ -pAPS. Note, that in this strategy the initial  $\alpha_1$  and  $\alpha_2$  are computed according to (11.2). As the second scalar parameter selection method we utilize the approach presented in [27] for automatically computing scalar  $\lambda$ 's whereby  $\alpha_1$  and  $\alpha_2$  are fixed and chosen according to (11.2). We refer to it as the  $\lambda$ -pAPS. In this approach we relate (11.1) to

## II. Parameter Selection Methods for Total Variation Models



**Figure 11.3:** Restoration via the LATV-algorithm (first two rows) and via the pLATV-algorithm (last two rows) using approach 2 of the cookies-image corrupted with Gaussian white noise with  $\sigma = \sqrt{0.02}$  (left column),  $\sigma = 0.1$  (middle column),  $\sigma = 0.01$  (right column) and salt-and-pepper noise.

the constrained optimization problem

$$\min_{u \in BV(\Omega)} \int_{\Omega} |Du| \quad \text{s.t.} \quad \frac{1}{|\Omega|} \mathcal{H}(u) = \mathcal{B}$$

where  $\mathcal{H}(u) := \alpha_1 \|T_1 u - g_1\|_{L^1(\Omega)} + \alpha_2 \|T_2 u - g_2\|_{L^2(\Omega)}$ ,  $\mathcal{B} = \alpha_1 \nu_1 + \alpha_2 \nu_2$  is a constant depending on the noise, and  $|\Omega|$  denotes the volume of  $\Omega$ . Then the so-called pAPS-algorithm proposed in [27] is utilized to compute suitable parameters  $\lambda$ .

## 11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise

image	$\sigma$	$s$	$\alpha_1\text{-}\alpha_2\text{-pAPS}$		$\lambda\text{-pAPS}$		LATV approach 1 with $\omega = 19$		pLATV approach 1 with $\omega = 3$	
			PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM
phantom	$\sqrt{0.02}$	0.005	22.32	0.7641	22.73	0.7730	23.53	0.8137	23.27	0.8044
		0.01	21.08	0.6752	21.84	0.6852	22.37	0.7211	22.00	0.6941
	0.1	0.005	23.99	0.8190	24.85	0.8348	26.01	0.8770	25.65	0.8670
		0.01	22.85	0.7479	24.63	0.7975	25.31	0.8388	25.27	0.8428
	0.01	0.005	28.14	0.9747	27.63	0.9704	31.61	0.9908	30.07	0.9868
		0.01	25.54	0.9555	25.23	0.9510	28.56	0.9854	27.69	0.9760
	cameraman	$\sqrt{0.02}$	0.005	23.21	0.6252	24.33	0.6625	23.91	0.6683	23.92
		0.01	22.38	0.5520	24.24	0.6181	23.22	0.5945	23.24	0.5894
		0.1	0.005	24.28	0.6652	26.00	0.7441	25.82	0.7542	25.77
		0.01	23.80	0.6195	25.78	0.7223	25.84	0.7348	25.78	0.7217
cookies	$\sqrt{0.02}$	0.005	32.42	0.9385	32.41	0.9388	33.01	0.9468	32.57	0.9466
		0.01	31.36	0.9325	31.73	0.9342	32.01	0.9436	31.31	0.9419
	0.1	0.005	23.65	0.7013	24.92	0.7624	24.32	0.7351	24.46	0.7399
		0.01	23.33	0.6586	25.04	0.7694	24.01	0.6948	24.14	0.7105
	0.01	0.005	25.25	0.7654	25.99	0.8215	26.20	0.8178	26.31	0.8280
		0.01	25.47	0.7773	26.13	0.8252	26.26	0.8188	26.45	0.8336
	0.01	0.005	33.03	0.9599	35.40	0.9734	36.09	0.9739	35.59	0.9737
		0.01	33.05	0.9603	35.02	0.9721	35.53	0.9724	35.48	0.9724

**Table 11.1:** Removal of Gaussian + salt-and-pepper noise.

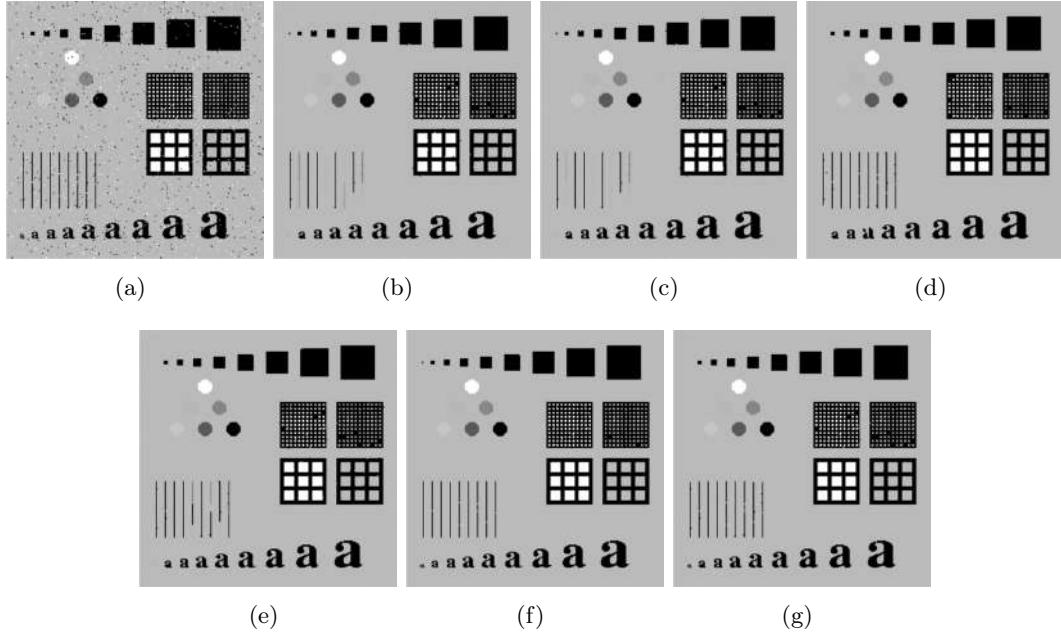
For our comparison we distort the original images shown in Figure 11.1 with mixed Gaussian-impulse noise with different noise-levels, i.e.,  $\sigma = \{\sqrt{0.02}, 0.1, 0.01\}$ ,  $s = \{0.01, 0.005\}$ , and  $r = \{0.01, 0.005\}$ , and restore them with the above mentioned algorithms. Our findings are summarized in Table 11.1 and Table 11.2 for Gaussian plus salt-and-pepper noise removal and in Table 11.3 and Table 11.4 for eliminating mixed Gaussian - random-valued noise. From these tables we observe that the LATV- and pLATV-algorithm using approach 2 for the local bounds outperform (with respect to the quality measures PSNR and MSSIM) the scalar parameter selection strategies in all considered experiments. This is slightly different for using approach 1 in the LATV- and pLATV-algorithm. While for removing Gaussian + random-valued noise these algorithms still give better results in all experiments than the scalar parameter methods, this is not always the case for the simultaneous removal of Gaussian and salt-and-pepper noise. In the latter case in a few cases the scalar parameter algorithms generate results with larger PSNR and MSSIM.

From all these experiments we observe that the LATV-algorithm using approach 2 for the local bounds seems to give always the results with the largest PSNR and MSSIM. This clearly indicates that the choice of the local bounds are very crucial in order to utilize the strength of locally adaptive parameters. The strength of these approaches is also visible in Figure 11.4 - 11.8, where we depict for certain noise-levels the generated reconstruction of the corresponding algorithm. For example in Figure 11.4 we observe that the scalar parameter choice methods are not able to preserve all features. There, e.g., the bars in the left upper part are partly vanished. On the contrary, the locally adaptive parameter choice methods are able to retain these features much better. In particular, the LATV-algorithm with approach 2 gives the visually best results, which is also affirmed by the largest PSNR and MSSIM. The presented figures show, that the reconstructions of the locally adaptive algorithms depend on the local bounds and that they are able to generate visually much better results than scalar parameter methods.

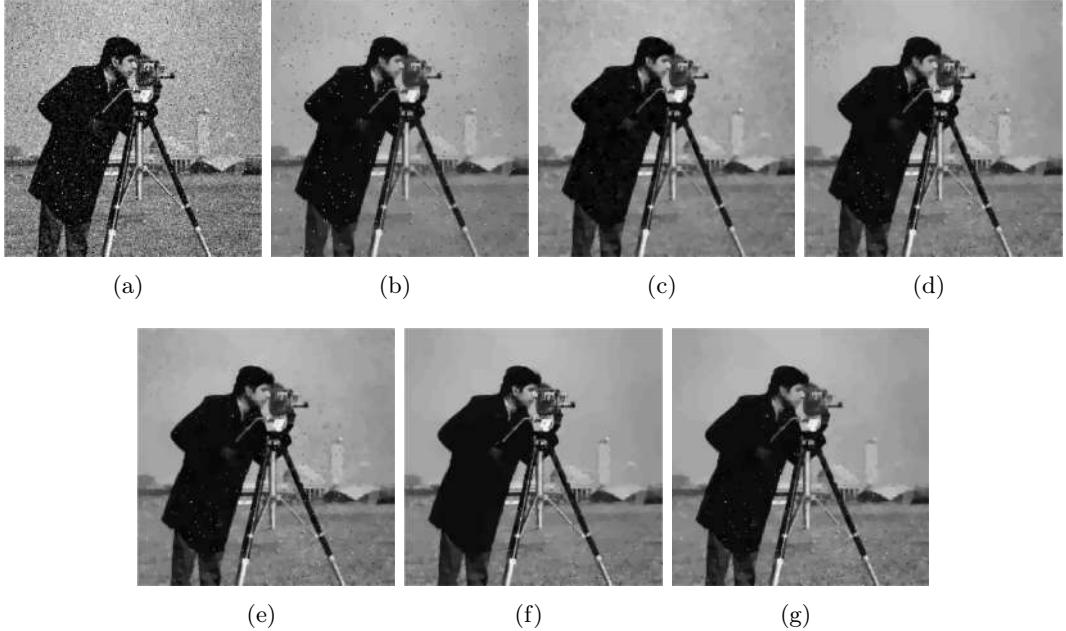
## II. Parameter Selection Methods for Total Variation Models

image	$\sigma$	s	LATV approach 2 with $\omega = 3$		pLATV approach 2 with $\omega = 3$	
			PSNR	MSSIM	PSNR	MSSIM
phantom	$\sqrt{0.02}$	0.005	25.77	0.9464	23.54	0.8608
		0.01	24.97	0.9431	23.05	0.8654
	0.1	0.005	27.94	0.9634	25.81	0.9094
		0.01	27.03	0.9627	25.23	0.9166
	0.01	0.005	35.96	0.9957	32.27	0.9918
		0.01	32.89	0.9950	29.80	0.9885
cameraman	$\sqrt{0.02}$	0.005	25.78	0.7908	24.88	0.7585
		0.01	25.64	0.7932	24.47	0.7381
	0.1	0.005	27.34	0.8299	25.81	0.7808
		0.01	27.27	0.8298	26.05	0.7926
	0.01	0.005	36.69	0.9607	32.91	0.9499
		0.01	35.49	0.9598	31.60	0.9420
cookies	$\sqrt{0.02}$	0.005	24.98	0.8032	24.42	0.7587
		0.01	24.89	0.8008	24.54	0.7761
	0.1	0.005	26.62	0.8481	26.25	0.8292
		0.01	26.37	0.8413	26.00	0.8263
	0.01	0.005	38.16	0.9772	36.50	0.9774
		0.01	37.77	0.9793	36.06	0.9763

**Table 11.2:** Removal of Gaussian + salt-and-pepper noise via the LATV- and pLATV-algorithm using approach 2 for the local bounds.



**Figure 11.4:** (a) Noisy image with  $\sigma = 0.01$  and  $s = 0.01$ ; (b) Restoration via  $\alpha_1\text{-}\alpha_2\text{-}pAPS$ ; (c) Restoration via  $\lambda\text{-}pAPS$ ; (d) Restoration via LATV approach 1; (e) Restoration via pLATV approach 1; (f) Restoration via LATV approach 2; (g) Restoration via pLATV approach 2.



**Figure 11.5:** (a) Noisy image with  $\sigma = 0.1$  and  $s = 0.005$ ; (b) Restoration via  $\alpha_1\text{-}\alpha_2\text{-}pAPS$ ; (c) Restoration via  $\lambda\text{-}pAPS$ ; (d) Restoration via LATV approach 1; (e) Restoration via pLATV approach 1; (f) Restoration via LATV approach 2; (g) Restoration via pLATV approach 2.

### 11.3.3. Separated Noise

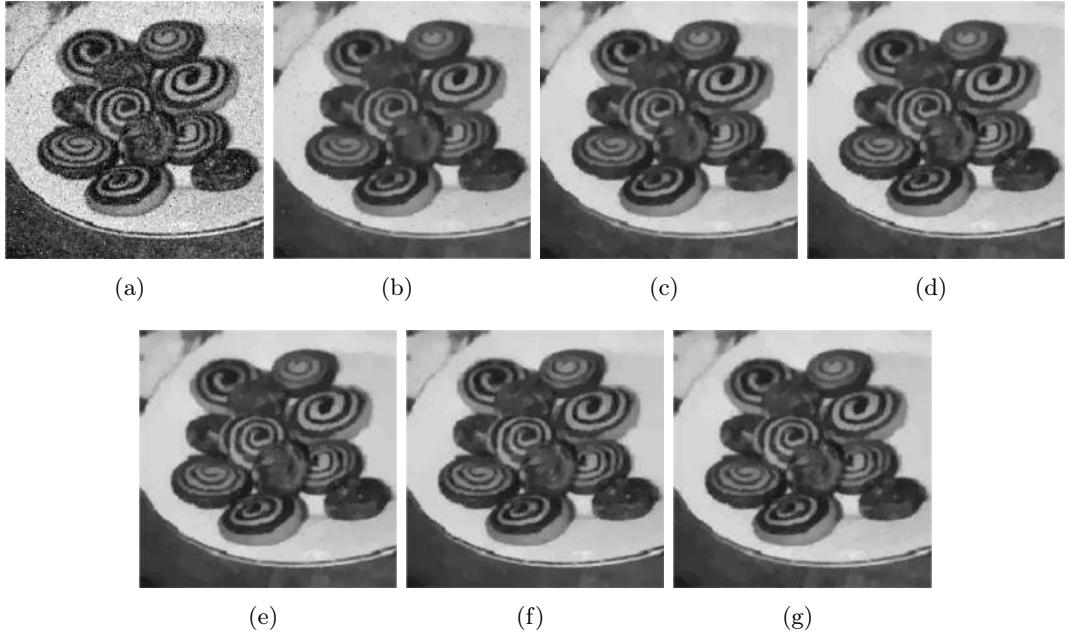
Here we consider an example where salt-and-pepper noise and Gaussian noise is disjoint present in an image. More precisely, we consider the image in Figure 11.9(a) where the lower half  $g_1$  is contaminated only with salt-and-pepper noise ( $s = 0.1$ ) and in the upper half  $g_2$  only Gaussian white noise ( $\sigma = 0.1$ ) is contained. That is, in this situation  $g_1 \neq g_2$ . The associated image domains are  $\Omega_1$  and  $\Omega_2$ , which are two disjoint subdomains of  $\Omega$  such that  $\Omega_1 = \Omega \setminus \Omega_2$  and  $\Omega = \Omega_1 \cup \Omega_2$ . We are aware that this is a rather artificial example. However, it is very interesting from a numerical point of view, since a correct global solution cannot be obtained by just separating the image into an upper and a lower part due to the nonadditivity of the total variation [20]. We set  $T_1 = \chi_{\Omega_1}$  and  $T_2 = \chi_{\Omega_2}$  in (11.6). Since there is only one type of noise in  $\Omega_1$  and only one type of noise in  $\Omega_2$ , which is different to the noise in  $\Omega_1$ , the bound  $B = \alpha_1\nu_1 + \alpha_2\nu_2$  in (11.7) does not seem reasonable anymore for all  $x \in \Omega$ . In particular, for approach 1 a better choice seems to choose  $B$  also spatially varying but not fully local as in approach 2. That is, in approach 1 we replace  $B(u)$  by

$$B(u)(x) := \alpha_1\chi_{\Omega_1}(x)\nu_1(u) + \alpha_2\chi_{\Omega_2}(x)\nu_2(u),$$

where  $\nu_1(u)$  is the EAV of salt-and-pepper noise and  $\nu_2(u)$  is the variance of Gaussian noise, in our proposed locally adaptive total variation algorithms. For approach 2 we analogously set

$$B(u)(x) := c(\alpha_1\chi_{\Omega_1}(x)\nu_1(u)(x) + \alpha_2\chi_{\Omega_2}(x)\nu_2(u)(x)).$$

## II. Parameter Selection Methods for Total Variation Models



**Figure 11.6:** (a) Noisy image with  $\sigma = 0.1$  and  $s = 0.005$ ; (b) Restoration via  $\alpha_1$ - $\alpha_2$ -pAPS; (c) Restoration via  $\lambda$ -pAPS; (d) Restoration via LATV; (e) Restoration via pLATV; (f) Restoration via LATV approach 2; (g) Restoration via pLATV approach 2.

In [28] it is demonstrated by using the  $\alpha_1$ - $\alpha_2$ -pAPS, which chooses suitable parameters  $\alpha_1$  and  $\alpha_2$ , that the  $L^1$ - $L^2$ -TV model is able to remove both types of noise considerably while preserving details at the same time from such images; see Figure 11.9(b). Using locally adaptive parameters this restoration capability can be improved; see Figure 11.9. In particular the pLATV-algorithm with approach 1 generates the best restoration with respect to PSNR, while the restoration obtained via the LATV-algorithm with approach 2 yields the largest MSSIM.

The  $\lambda$ -pAPS algorithm is clearly not suited to this task (see Figure 11.9(c)), since  $\alpha_1$  and  $\alpha_2$  are kept unchanged and  $\lambda$  is globally constant which makes it impossible to weight certain local parts of the image differently.

### 11.3.4. Nonhomogeneous Noise

We consider now an example, where the noise is nonhomogeneously distributed over the image-domain. In particular, we look at the cameraman image degraded by salt-and-pepper noise with  $s = 0.1$  and Gaussian white noise with variance  $\sigma^2 = 0.01$  in the whole domain  $\Omega$  except a rather small area  $\tilde{\Omega}$  (highlighted in red in Figure 11.10(a)), where the variance of the Gaussian noise is 6 times larger, i.e., the variance  $\tilde{\sigma}^2 = 0.06$  in this part. Since the noise is locally varying in this application, the bound  $B$  in approach 1 has to be adjusted to this situation. This is done by making  $\nu_\tau$ ,  $\tau = 1, 2$ , locally dependent and we write  $\nu_\tau = \nu_\tau(\hat{u})(x)$  to stress the dependency on the true image  $\hat{u}$  and on the localization  $x \in \Omega$  in the image. Then  $B(u)(x) = \alpha_1\nu_1(u)(x) + \alpha_2\nu_2(u)(x)$ .

The results obtained by the  $\lambda$ -pAPS and LATV-algorithm using approach 1 and approach

## 11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise

image	$\sigma$	$r$	$\alpha_1\text{-}\alpha_2\text{-pAPS}$		$\lambda\text{-pAPS}$		LATV approach 1 with $\omega = 11$		pLATV approach 1 with $\omega = 11$	
			PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM
phantom	$\sqrt{0.02}$	0.005	24.84	0.8988	24.92	0.8958	25.88	0.9234	24.99	0.8988
		0.01	24.13	0.8726	24.28	0.8701	25.32	0.9111	24.35	0.8738
	0.1	0.005	26.94	0.9228	26.94	0.9250	28.08	0.9417	27.02	0.9271
		0.01	25.94	0.8991	26.08	0.9092	27.42	0.9400	26.57	0.9245
	0.01	0.005	33.69	0.9919	31.61	0.9872	34.79	0.9929	33.08	0.9907
		0.01	30.55	0.9858	28.20	0.9725	32.43	0.9885	28.21	0.9726
cameraman	$\sqrt{0.02}$	0.005	25.13	0.7446	25.28	0.7465	25.65	0.7712	25.43	0.7619
		0.01	24.86	0.7271	25.15	0.7404	25.48	0.7609	25.27	0.7511
	0.1	0.005	26.35	0.7715	26.43	0.7792	27.00	0.8074	26.68	0.7943
		0.01	25.99	0.7533	26.13	0.7779	26.82	0.8027	26.32	0.7746
	0.01	0.005	33.39	0.9414	33.10	0.9412	35.52	0.9539	34.10	0.9520
		0.01	32.58	0.9383	32.58	0.9383	34.39	0.9496	32.53	0.9391
cookies	$\sqrt{0.02}$	0.005	24.27	0.7564	24.67	0.7724	24.74	0.7806	24.76	0.7780
		0.01	24.12	0.7475	24.63	0.7740	24.69	0.7758	24.64	0.7659
	0.1	0.005	25.65	0.8009	25.97	0.8191	26.14	0.8236	26.08	0.8164
		0.01	25.48	0.7960	25.71	0.8144	25.99	0.8208	25.95	0.8175
	0.01	0.005	36.01	0.9756	36.03	0.9758	37.21	0.9758	36.19	0.9759
		0.01	34.64	0.9692	35.60	0.9741	36.57	0.9740	35.74	0.9742

Table 11.3: Removal of Gaussian + random-valued impulse noise.

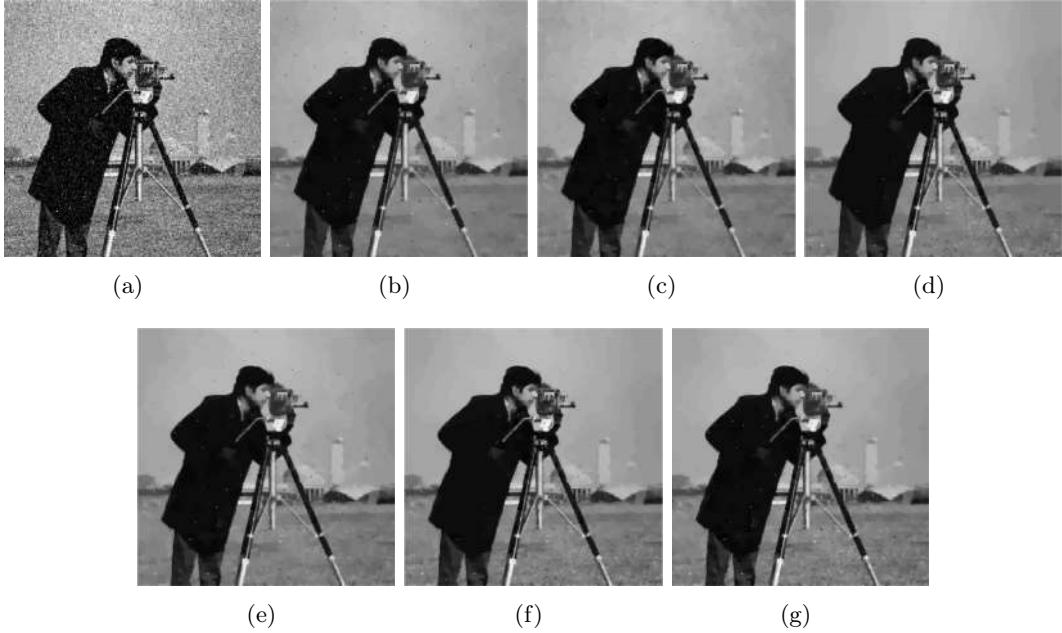


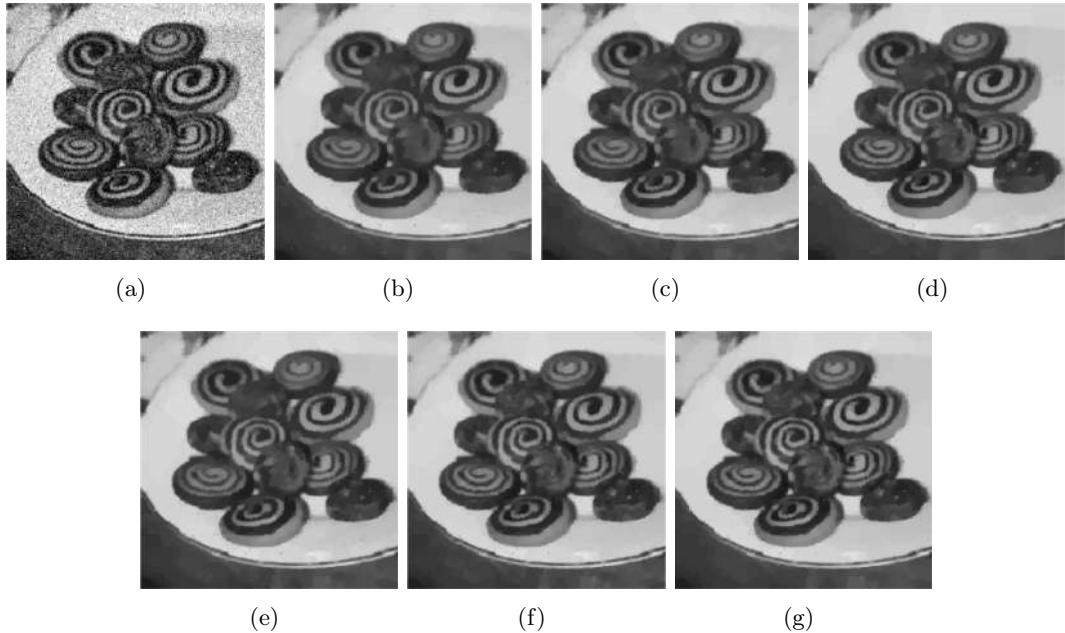
Figure 11.7: (a) Noisy image with  $\sigma = 0.1$  and  $r = 0.005$ ; (b) Restoration via  $\alpha_1\text{-}\alpha_2\text{-pAPS}$ ; (c) Restoration via  $\lambda\text{-pAPS}$ ; (d) Restoration via LATV approach 1; (e) Restoration via pLATV approach 1; (f) Restoration via LATV approach 2; (g) Restoration via pLATV approach 2

2 are shown in Figure 11.10. We observe that the  $\lambda$ -pAPS-algorithm is neither able to remove the noise in  $\tilde{\Omega}$  nor in the rest of the image. The LATV-algorithm with approach 1 is able to remove the noise in  $\tilde{\Omega}$  considerably, but not the outliers in the rest of the image. A reason for this may be, that in this approach the values  $\nu_1$  and  $\nu_2$  may not be good approximations

## II. Parameter Selection Methods for Total Variation Models

image	$\sigma$	$s$	LATV approach 2 with $\omega = 3$		pLATV approach 2 with $\omega = 3$	
			PSNR	MSSIM	PSNR	MSSIM
phantom	$\sqrt{0.02}$	0.005	27.15	0.9555	26.05	0.9306
		0.01	26.57	0.9519	24.87	0.9001
	0.1	0.005	29.49	0.9678	28.79	0.9633
		0.01	28.74	0.9661	26.94	0.9309
	0.01	0.005	38.67	0.9973	35.30	0.9944
		0.01	35.74	0.9932	33.70	0.9931
cameraman	$\sqrt{0.02}$	0.005	26.01	0.7892	25.58	0.7761
		0.01	25.83	0.7826	25.38	0.7681
	0.1	0.005	27.39	0.8235	26.87	0.8092
		0.01	27.19	0.8191	26.67	0.8055
	0.01	0.005	36.00	0.9574	34.54	0.9559
		0.01	35.04	0.9549	33.84	0.9528
cookies	$\sqrt{0.02}$	0.005	25.25	0.8016	25.16	0.7995
		0.01	25.14	0.7984	24.95	0.7835
	0.1	0.005	26.71	0.8442	26.62	0.8418
		0.01	26.56	0.8425	26.51	0.8408
	0.01	0.005	37.72	0.9777	36.74	0.9779
		0.01	37.56	0.9782	36.47	0.9771

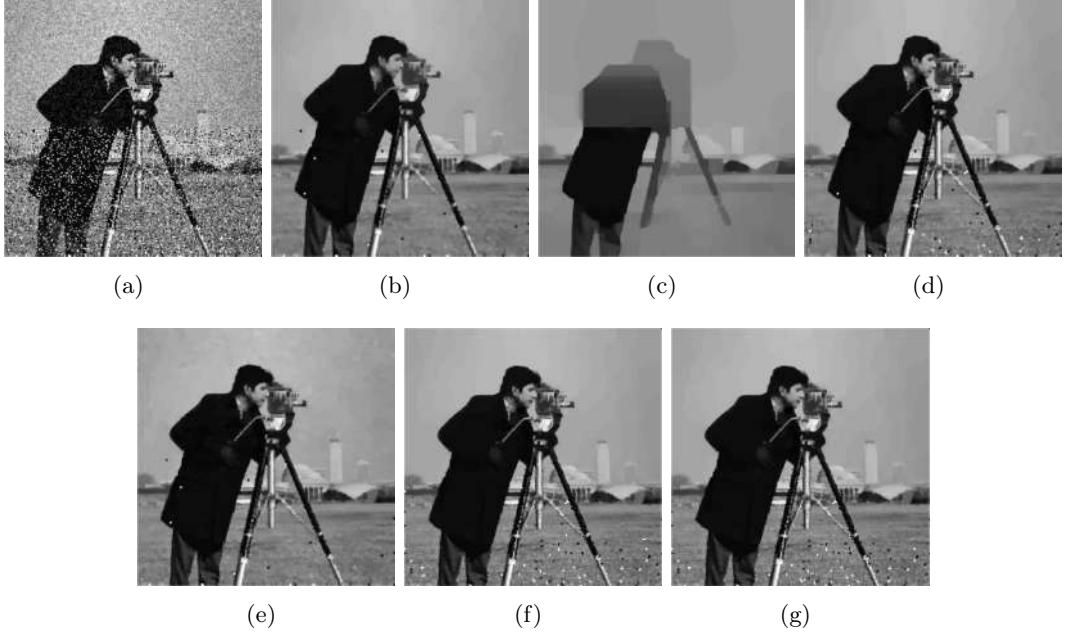
**Table 11.4:** Removal of Gaussian + random-valued noise via the LATV- and pLATV-algorithm using approach 2.



**Figure 11.8:** (a) Noisy image with  $\sigma = 0.1$  and  $r = 0.005$ ; (b) Restoration via  $\alpha_1\text{-}\alpha_2\text{-}pAPS$ ; (c) Restoration via  $\lambda\text{-}pAPS$ ; (d) Restoration via LATV approach 1; (e) Restoration via pLATV approach 1; (f) Restoration via LATV approach 2; (g) Restoration via pLATV approach 2.

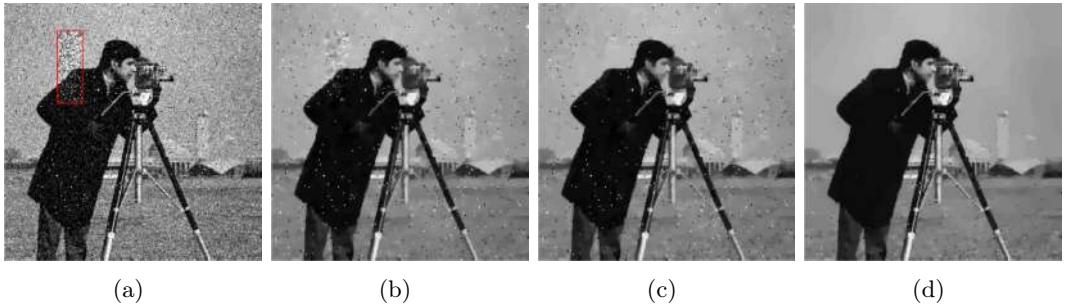
of the true values leading to an unreliable bound  $B = \sum_{x \in \Omega} B(u)(x)$ . On the contrary, if we use approach 2 and hence the true variance and EAV, then the LATV-algorithm generates a

## 11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise



**Figure 11.9:** (a) Noisy image with  $s = 0.1$  and  $\sigma = 0.1$ ; (b) Restoration via  $\alpha_1\text{-}\alpha_2\text{-}pAPS$  (PSNR: 25.97; MSSIM: 0.8477); (c) Restoration via  $\lambda\text{-}pAPS$  (PSNR: 18.09; MSSIM: 0.6414); (d) Restoration via LATV approach 1 (PSNR: 25.74; MSSIM: 0.8609); (e) Restoration via pLATV approach 1 (PSNR: 26.41; MSSIM: 0.8609); (f) Restoration via LATV approach 2 (PSNR: 25.34; MSSIM: 0.8687); (g) Restoration via pLATV approach 2 (PSNR: 25.26; MSSIM: 0.8665).

very satisfactory result, where all the noise seems to be removed, see Figure 11.10(d). This once more demonstrates that the choice of the local bounds is very crucial for the success of a locally varying regularization parameter method.



**Figure 11.10:** (a) Noisy image with  $s = 0.005$ ,  $\sigma = 0.1$ , and  $\sigma = \sqrt{0.06}$  in the highlighted area; (b) Restoration via  $\lambda\text{-}pAPS$  (PSNR: 22.33; MSSIM: 0.5886); (c) Restoration via LATV approach 1 (PSNR: 23.00; MSSIM: 0.6246); (d) Restoration via LATV approach 2 (PSNR: 26.24; MSSIM: 0.8065).

## II. Parameter Selection Methods for Total Variation Models

### 11.4. References

- [1] M. Alkämper and A. Langer. Using DUNE-ACFem for non-smooth minimization of bounded variation functions. *Archive of Numerical Software*, 5(1):3–19, 2017.
- [2] A. Almansa, C. Ballester, V. Caselles, and G. Haro. A TV based restoration model with local constraints. *Journal of Scientific Computing*, 34(3):209–236, 2008.
- [3] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, 2000.
- [4] M. Bertalmío, V. Caselles, B. Roug  , and A. Sol  . TV based image restoration with local constraints. *Journal of Scientific Computing*, 19(1-3):95–122, 2003.
- [5] A. C. Bovik. *Handbook of Image and Video Processing*. Academic press, 2010.
- [6] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.
- [7] J.-F. Cai, R. H. Chan, and M. Nikolova. Two-phase approach for deblurring images corrupted by impulse plus Gaussian noise. *Inverse Problems and Imaging*, 2(2):187–204, 2008.
- [8] L. Calatroni, J. C. D. L. Reyes, and C.-B. Sch  nlieb. Infimal convolution of data discrepancies for mixed noise removal. *SIAM Journal on Imaging Sciences*, 10(3):1196–1233, 2017.
- [9] V. C. Cao, J. C. D. L. Reyes, and C.-B. Sch  nlieb. Learning optimal spatially-dependent regularization parameters in total variation image restoration. *Inverse Problems*, 33(7):074005, 2017.
- [10] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [11] D.-Q. Chen and L.-Z. Cheng. Spatially adapted regularization parameter selection based on the local discrepancy function for Poissonian image deblurring. *Inverse Problems*, 28(1):015004, 2011.
- [12] J. Delon and A. Desolneux. A patch-based approach for removing impulse or mixed Gaussian-impulse noise. *SIAM Journal on Imaging Sciences*, 6(2):1140–1174, 2013.
- [13] B. Dong, H. Ji, J. Li, Z. Shen, and Y. Xu. Wavelet frame based blind image inpainting. *Applied and Computational Harmonic Analysis*, 32(2):268–279, 2012.
- [14] Y. Dong, M. Hinterm  ller, and M. M. Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *Journal of Mathematical Imaging and Vision*, 40(1):82–104, 2011.
- [15] R. Garnett, T. Huegerich, C. Chui, and W. He. A universal noise removal algorithm with an impulse detector. *IEEE Transactions on Image Processing*, 14(11):1747–1754, 2005.
- [16] O. Ghita and P. F. Whelan. A new GVF-based image enhancement formulation for use in the presence of mixed noise. *Pattern Recognition*, 43(8):2646–2658, 2010.
- [17] G. Gilboa, N. Sochen, and Y. Y. Zeevi. Texture preserving variational denoising using an adaptive fidelity term. In *Proc. VLsM*, volume 3, 2003.
- [18] E. Giusti. *Minimal Surfaces and Functions of Bounded Variation*, volume 80 of *Monographs in Mathematics*. Birkh  user Verlag, Basel, 1984.
- [19] Z. Gong, Z. Shen, and K.-C. Toh. Image restoration with mixed or unknown noises. *Multiscale Modeling & Simulation*, 12(2):458–487, 2014.
- [20] M. Hinterm  ller and A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1/L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.

## 11. Locally Adaptive Total Variation for Removing Mixed Gaussian-Impulse Noise

- [21] M. Hintermüller and A. Langer. Adaptive regularization for Parseval frames in image processing. *SFB-Report No. 2014-014*, page 12, 2014.
- [22] M. Hintermüller and A. Langer. Non-overlapping domain decomposition methods for dual total variation based image denoising. *Journal of Scientific Computing*, 62(2):456–481, 2015.
- [23] M. Hintermüller and C. N. Rautenberg. Optimal selection of the regularization function in a weighted total variation model. Part I: Modelling and theory. *Journal of Mathematical Imaging and Vision*, pages 1–17, 2017.
- [24] M. Hintermüller, C. N. Rautenberg, T. Wu, and A. Langer. Optimal selection of the regularization function in a weighted total variation model. Part II: Algorithm, its analysis and numerical tests. *Journal of Mathematical Imaging and Vision*, pages 1–19, 2017.
- [25] M. Hintermüller and M. M. Rincon-Camacho. Expected absolute value estimators for a spatially adapted regularization parameter choice rule in  $L^1$ -TV-based image restoration. *Inverse Problems*, 26(8):085005, 30, 2010.
- [26] Y.-M. Huang, M. K. Ng, and Y.-W. Wen. Fast image restoration methods for impulse and Gaussian noises removal. *IEEE Signal Processing Letters*, 16(6):457–460, 2009.
- [27] A. Langer. Automated parameter selection for total variation minimization in image restoration. *Journal of Mathematical Imaging and Vision*, 57(2):239–268, 2017.
- [28] A. Langer. Automated parameter selection in the  $L^1$ - $L^2$ -TV model for removing Gaussian plus impulse noise. *Inverse Problems*, 33(7):074002, 2017.
- [29] B. Li, Q. Liu, J. Xu, and X. Luo. A new method for removing mixed noises. *Science China Information Sciences*, 54(1):51–59, 2011.
- [30] F. Li, M. K. Ng, and C. Shen. Multiplicative noise removal with spatially varying regularization parameters. *SIAM Journal on Imaging Sciences*, 3(1):1–20, 2010.
- [31] F. Li and T. Zeng. Image restoration via tight frame regularization and local constraints. *Journal of Scientific Computing*, 57(2):349–371, 2013.
- [32] Y.-R. Li, L. Shen, D.-Q. Dai, and B. W. Suter. Framelet algorithms for de-blurring images corrupted by impulse plus Gaussian noise. *IEEE Transactions on Image Processing*, 20(7):1822–1837, 2011.
- [33] J. Liu, X.-C. Tai, H. Huang, and Z. Huan. A weighted dictionary learning model for denoising images corrupted by mixed noise. *IEEE Transactions on Image Processing*, 22(3):1108–1120, 2013.
- [34] R. W. Liu, L. Shi, S. C. H. Yu, and D. Wang. Box-constrained second-order total generalized variation minimization with a combined  $L^{1,2}$  data-fidelity term for image reconstruction. *Journal of Electronic Imaging*, 24(3):033026–033026, 2015.
- [35] E. López-Rubio. Restoration of images corrupted by Gaussian and uniform impulsive noise. *Pattern Recognition*, 43(5):1835–1846, 2010.
- [36] S. Peng and L. Lucke. Fuzzy filtering for mixed noise removal during image processing. In *Proceedings of the Third IEEE Conference on Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence*, pages 89–93. IEEE, 1994.
- [37] P. Rodríguez, R. Rojas, and B. Wohlberg. Mixed gaussian-impulse noise image restoration via total variation. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pages 1077–1080. IEEE, 2012.
- [38] Y. Shen, B. Han, and E. Braverman. Removal of mixed Gaussian and impulse noise using directional tensor product complex tight framelets. *Journal of Mathematical Imaging and Vision*, pages 1–14, 2015.

## **II. Parameter Selection Methods for Total Variation Models**

- [39] D. M. Strong, P. Blomgren, and T. F. Chan. Spatially adaptive local-feature-driven total variation minimizing image restoration. In *Optical Science, Engineering and Instrumentation'97*, pages 222–233. International Society for Optics and Photonics, 1997.
- [40] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
- [41] Y. Xiao, T. Zeng, J. Yu, and M. K. Ng. Restoration of images corrupted by mixed Gaussian-impulse noise via  $l_1 - l_0$  minimization. *Pattern Recognition*, 44(8):1708–1720, 2011.
- [42] M. Yan. Restoration of images corrupted by impulse noise and mixed Gaussian impulse noise using blind inpainting. *SIAM Journal on Imaging Sciences*, 6(3):1227–1245, 2013.
- [43] J. X. Yang and H. R. Wu. Mixed Gaussian and uniform impulse noise analysis using robust estimation for digital images. In *16th International Conference on Digital Signal Processing, 2009*, pages 1–5. IEEE, 2009.

## Part III.

# Finite Element Discretization of the Total Variation



## 12. Introduction and Overview

The world around us and images in the environment, before being sensed, are naturally analog, which means that the image exists on a continuous domain and takes values that come from a continuum of possibilities [8]. Due to digitalization of images, the image information is usually stored as a set of pixels in a row-column format. Depending on the pixel size (scale of the digital image) features of the original analog image may be lost in the digital one. Sufficient sampling, such that no important information in the digital image is lost, may result into large-scale data sets that have to be further processed (e.g. often noise contained in digital images should be eliminated). In order to address extremely large problems in real-time, subspace correction and domain decomposition methods, as described in Part I, are needed, allowing us to split the computational workload. While these techniques allow the post-processing of large-scale images, they do not reduce the number of pixels of an image.

The pixel-wise storage of an image in a column-row format promotes the usually used finite difference discretisation of the considered image reconstruction problem. However, we note that besides the finite difference scheme for solving partial differential equations finite element methods (FEMs) have been widely used, due to their flexibility when dealing with complex geometries and complicated boundary conditions. An additional advantage of FEMs is, that they allow relatively easy for an adaptive discretization [19]. In order to utilize a FEM a weak formulation of the considered optimisation problem is required. For the inverse problem of reconstructing an image from observed corrupted data one usually minimizes a nonsmooth functional, which does not allow usual derivatives and makes it rather difficult to find its weak formulation. This might be a further reason for mainly using finite differences in solving imaging problems. However, there exist several good reasons to consider finite element discretizations for imaging applications, and hence study FEMs for total variation minimization problems. For example, a finite element discretization is needed, when the image data is not represented in a Cartesian grid, which occur, e.g., due to the use of acquisition equipment with hexagonal sensor layouts [11, 15]. Further, a finite element framework allows to reconstruct images defined on a manifold, e.g., in geodesy [14], and is applicable in medical imaging applications [41].

First attempts for using FEMs in image restoration have been made for nonlinear diffusion models, which are used mainly for image denoising and edge detection. In particular, in [2, 6, 20] adaptive FEMs are proposed for the modified Perona-Malik model introduced in [9], and in [17] an optimal error estimate for a finite element discretization of this diffusion model is derived.

Based on these considerations FEMs for smoothed versions of model (1.1), where the total variation term is either replaced by a Tikhonov regularizer [16] or a smoothed total variation [21], are introduced and investigated for image denoising. In [16] even the removal of mixed Gaussian-impulse noise is considered. Later, directly the  $L^2$ -TV model with  $T = I$  is considered with respect to a finite element discretization [3, 4], where the optimization problem is formulated as a saddle point problem. In [3] with  $T = I$ , it is investigated for which finite element spaces a solution of the discretized  $L^2$ -TV model converges to the solution of the

### III. Finite Element Discretization of the Total Variation

continuous  $L^2$ -TV model as the mesh-size tends to zero. This convergence is however only shown for piecewise affine and globally continuous FE functions. For piecewise constant functions even a counterexample can be constructed, disproving the convergence to the continuous solution for such functions in general.

Very recently in [12] a novel discrete total variation based on a nodal quadrature formula is defined. This variant possesses a convenient dual representation, which allows to use a variety of algorithms for image reconstruction problems including the  $L^2$ -TV and  $L^1$ -TV model.

In order to diminish the dimensionality of images without losing important information of the analog image, a good way would be to discretize the image in an adaptive way by taking features of the pictured environment into account. Note that images usually have homogeneous regions as well as parts with a lot of details. Hence, it seems intuitively sufficient to discretize fine in regions of transitions and small details, while a coarse discretization should do well in uniform parts. In this vein, based on the primal-dual method of [10] in [5] an adaptive FEM using a posteriori error estimates for model (1.1) with  $T = I$  is proposed. Note, that in [5] only clean images and no image reconstruction problem are considered. In a similar way relying on a semismooth Newton method for solving a smoothed version of model (1.1) in [13] an adaptive FEM, again using a posteriori error estimates, is suggested for image denoising.

In this part, consisting of Chapter 13, which results are published in [AL11], we present an efficient implementation of the  $L^1$ - $L^2$ -TV model with  $T_1 = T_2 = I$ . In particular we consider a finite element discretization on conforming and locally adaptive meshes. We prove that a minimizer of the treated model on a suitable finite element space indeed converges to a minimizer of the original problem in the space of bounded variation functions. This convergence holds, as in [3], only for the finite element space of piecewise affine and globally continuous functions. In order to obtain a solution of the considered optimisation problem, we adapt the primal-dual algorithm of [10] and implement it in the programming environment DUNE [7]. By using DUNE-ALUGRID [1] and thanks to the ability of DUNE-ACFEM the resulting algorithm is intrinsically parallelized by domain decomposition. Further, the implementation is based on an adaptive refinement of the mesh, where the refinement is performed heuristically at known discontinuities. This demonstrates that an efficient implementation of nonsmooth total variation minimization problems in a FE framework is indeed possible.

## References

- [1] M. Alkämper, A. Dedner, R. Klöfkorn, and M. Nolte. The dune-alugrid module. *Archive of Numerical Software*, 4(1):1–28, 2016.
- [2] E. Bänsch and K. Mikula. A coarsening finite element strategy in image selective smoothing. *Computing and Visualization in Science*, 1(1):53–61, 1997.
- [3] S. Bartels. Total variation minimization with finite elements: convergence and iterative solution. *SIAM Journal on Numerical Analysis*, 50(3):1162–1180, 2012.
- [4] S. Bartels. Broken Sobolev space iteration for total variation regularized minimization problems. *IMA Journal of Numerical Analysis*, 36(2):493–502, 2015.
- [5] S. Bartels. Error control and adaptivity for a variational model problem defined on functions of bounded variation. *Mathematics of Computation*, 84(293):1217–1240, 2015.
- [6] C. Bazan and P. Blomgren. Adaptive finite element method for image processing. In *Proceedings of COMSOL Multiphysics Conference*, pages 377–381, 2005.

## 12. Introduction and Overview

- [7] M. Blatt, A. Burchardt, A. Dedner, C. Engwer, J. Fahlke, B. Flemisch, C. Gersbacher, C. Gräser, F. Gruber, C. Grüninger, D. Kempf, R. Klöfkorn, T. Malkmus, S. Müthing, M. Nolte, M. Piatakowski, and O. Sander. The distributed and unified numerics environment, version 2.4. *Archive of Numerical Software*, 4(100):13–29, 2016.
- [8] A. C. Bovik. *Handbook of Image and Video Processing*. Academic press, 2010.
- [9] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal on Numerical analysis*, 29(1):182–193, 1992.
- [10] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [11] S. Coleman, B. Scotney, and B. Gardiner. Tri-directional gradient operators for hexagonal image processing. *Journal of Visual Communication and Image Representation*, 38: 614–626, 2016.
- [12] M. Herrmann, R. Herzog, S. Schmidt, J. Vidal-Núñez, and G. Wachsmuth. Discrete Total Variation with Finite Elements and Applications to Imaging. *arXiv preprint arXiv:1804.07477*, 2018
- [13] M. Hintermüller and M. Rincon-Camacho. An adaptive finite element method in  $L^2$ -TV-based image denoising. *Inverse Problems and Imaging*, 8(3):685–711, 2014.
- [14] J. R. Jensen. *Introductory Digital Image Processing: A Remote Sensing Perspective*. Prentice-Hall Inc., Englewood Cliffs, N.J., 4th ed., 2015.
- [15] M. Knaup, S. Steckmann, O. Bockenbach, and M. Kachelrieß. CT image reconstruction using hexagonal grids. In 2007 IEEE Nuclear Science Symposium Conference Record, IEEE, 2007, pp. 3074–3076.
- [16] B. P. Lamichhane. Finite element techniques for removing the mixture of Gaussian and impulsive noise. *IEEE Transactions on Signal Processing*, 57(7):2538–2547, 2009.
- [17] J. Li. Finite element analysis for a nonlinear diffusion model in image processing. *Applied Mathematics Letters*, 15(2):197–202, 2002.
- [18] L. D. López Pérez. Regularisation of images defined on non-flat surfaces. PhD thesis, Université de Nice-Sophia Antipolis, 2006, <https://tel.archives-ouvertes.fr/tel-00141417v1>.
- [19] R. H. Nochetto, K. G. Siebert, and A. Veeraraghavan. Theory of adaptive finite element methods: an introduction. In: *DeVore R., Kunoth A. (eds.) Multiscale, Nonlinear and Adaptive Approximation*, 409–542, Springer, 2009.
- [20] T. Preußen and M. Rumpf. An adaptive finite element method for large scale image processing. *Journal of Visual Communication and Image Representation*, 11(2):183–195, 2000.
- [21] C. H. Yao. Finite element approximation for tv regularization. *International Journal of Numerical Analysis and Modeling.*, (3):516–526, 2008.



# 13. Using DUNE-ACFem for Nonsmooth Minimization of Bounded Variation Functions

**Bibliographic Note:** The content of this chapter is published in the article [AL11]: M. Alkämper and A. Langer. Using DUNE-ACFem for non-smooth minimization of bounded variation functions. *Archive of Numerical Software*, 5(1):3–19, 2017.

**Summary:** The utility of DUNE-ACFEM is demonstrated to work well for solving a nonsmooth minimization problem over bounded variation functions by implementing a primal-dual algorithm. The implementation is based on the simplification provided by DUNE-ACFEM. Moreover, the convergence of the discrete minimizer to the continuous one is shown theoretically.

## 13.1. Introduction

The DUNE-framework [7] and in particular the discretization module DUNE-FEM [11] provides the means to handle discrete functions, operators and solvers on different grids. Still, implementing complicated partial differential equations (PDEs) and their solvers is cumbersome and tedious. The DUNE-module DUNE-ACFEM aims to simplify the usage of DUNE-FEM by defining expression templates for discrete functions and PDE models. This allows to linearly combine discrete functions and PDE models. Additionally it supports parallel and adaptive finite-element schemes on continuous discrete functions for the predefined and combined models [14]. The flexibility of DUNE (and DUNE-FEM, DUNE-ACFEM) allows e.g. to exchange discrete spaces by a single line of code or to change the grid-type and linear solvers.

We will demonstrate the ease of implementation with DUNE-ACFEM by minimizing a nonsmooth functional consisting of a combined  $L^1/L^2$ -data fidelity term and a total variation term. Such an optimization problem has been shown to effectively remove Gaussian and salt-and-pepper noise, see [16, 19]. In order to compute an approximate solution we use the primal-dual algorithm proposed in [10], which requires a saddle point formulation of the problem. For the numerical implementation we discretize using finite-element spaces defined over locally refined conforming grids. Motivated by the works [3, 4, 5, 17], where the considered functional is composed solely of an  $L^2$ -data term and a total variation term, we refine the grid adaptively using an a priori criterion. Similar as in [3] we show for the considered minimization problem, that a minimizer over a finite element space converges to a minimizer in the space of functions of bounded variation as the mesh-size goes to 0.

In contrast to previous works [3, 4, 5, 17], we consider an additional nonsmooth  $L^1$ -data term in the objective, which has to be treated carefully. Moreover, due to the use of DUNE-ALUGRID [1] and the capabilities of DUNE-ACFEM the resulting algorithm is intrinsically parallelized by domain decomposition.

### III. Finite Element Discretization of the Total Variation

The rest of the paper is structured as follows. In Section 13.2 we formulate the continuous and the discrete problem with the respective discrete spaces. In particular for a certain discretization we prove that the discrete problem converges to the continuous one as the mesh-size goes to zero. In Section 13.3 we give a short overview of DUNE-ACFEM and discuss how it is used to implement the primal-dual algorithm. Numerical examples showing applicability of our proposed implementation and experiments testing different discretizations are presented in Section 13.4. Finally in Section 13.5 we conclude with a short summary and possible future research.

## 13.2. Problem Formulation

We consider the following problem

$$\min_{v \in BV(\Omega) \cap L^2(\Omega)} J_{\alpha_1, \alpha_2}(v) := \alpha_1 \|v - g\|_{L^1(\Omega)} + \alpha_2 \|v - g\|_{L^2(\Omega)}^2 + |Dv|(\Omega), \quad (13.1)$$

where  $\Omega \subset \mathbb{R}^d$ ,  $d \in \mathbb{N}$ , is an open bounded set with Lipschitz boundary,  $g \in L^2(\Omega)$  is a given datum,  $\alpha_i \geq 0$  for  $i = 1, 2$  with  $\alpha_1 + \alpha_2 > 0$  and  $BV(\Omega) \subset L^1(\Omega)$  denotes the space of functions with bounded variation. That is,  $v \in BV(\Omega)$  if and only if

$$|Dv|(\Omega) := \sup \left\{ \int_{\Omega} v \operatorname{div} \phi dx, \phi \in C_0^{\infty}(\Omega, \mathbb{R}^d), \|\phi\|_{C^0(\Omega)} \leq 1 \right\} \quad (13.2)$$

is finite, see [2, 13]. The space  $BV(\Omega)$  endowed with the norm  $\|v\|_{BV(\Omega)} = \|v\|_{L^1(\Omega)} + |Dv|(\Omega)$  is a Banach space [13]. For  $\alpha_2 > 0$  the minimization problem (13.1) admits a unique solution owing to the strict convexity of the quadratic term [19]. Note, that if  $\alpha_1 = 0$  in (13.1), then we obtain the functional used in [3, 4, 5, 17].

### 13.2.1. Discretization

Let  $(\mathcal{T}_h)_{h>0}$  be a sequence of shape-regular triangulations of the domain  $\Omega$  with diameter  $h = \max_{T \in \mathcal{T}_h} \operatorname{diam}(T)$  and  $\mathcal{S}_h$  be the set of its mesh entities of codimension 1 (i.e. edges for  $d = 2$ ). We define the following finite element spaces

$$\begin{aligned} \mathcal{L}^0(\mathcal{T}_h) &= \{q_h \in L^1(\Omega) : q_h|_T \text{ is constant for each } T \in \mathcal{T}_h\} \\ \mathcal{S}^1(\mathcal{T}_h) &= \{v_h \in C(\overline{\Omega}) : v_h|_T \text{ is affine for each } T \in \mathcal{T}_h\}. \end{aligned}$$

For the vector-valued versions, we write  $\mathcal{L}^0(\mathcal{T}_h)^d$  and  $\mathcal{S}^1(\mathcal{T}_h)^d$ , respectively, and boundary conditions are denoted via a subindex. In particular we use the subindex  $_0$  for zero boundary values and the subindex  $_N$  for zero boundary values in normal direction, e.g.,  $\mathcal{S}_0^1(\mathcal{T}_h) := \{v_h \in \mathcal{S}^1(\mathcal{T}_h) : v_h = 0 \text{ on } \partial\Omega\}$  and  $\mathcal{L}_N^0(\mathcal{T}_h)^d = \{q_h \in \mathcal{L}^0(\mathcal{T}_h)^d : q_h \cdot \mathbf{n} = 0 \text{ on } \partial\Omega\}$  where  $\mathbf{n}$  is the unit vector in normal direction. The nodal interpolant  $\mathcal{I}_h v \in \mathcal{S}^1(\mathcal{T}_h)$  of a function  $v \in W^{2,p}$ , with  $\frac{d}{2} < p \leq \infty$  or  $p = 1$  if  $d = 2$ , satisfies

$$\|v - \mathcal{I}_h v\|_{L^p(\Omega)} + h \|\nabla(v - \mathcal{I}_h v)\|_{L^p(\Omega)} \leq c_{\mathcal{I}} h^2 \|D^2 v\|_{L^p(\Omega)},$$

where  $c_{\mathcal{I}} > 0$  is a constant independent of  $h$ ; cf. [8].

We recall, that the space  $BV(\Omega)$  is continuously embedded in  $L^p(\Omega)$  for  $1 \leq p \leq \frac{d}{d-1}$ , i.e., there is a constant  $c_{BV} > 0$  such that  $\|v\|_{L^p(\Omega)} \leq c_{BV} \|v\|_{BV} = c_{BV} (\|v\|_{L^1(\Omega)} + |Dv|(\Omega))$  for

### 13. Using DUNE-ACFem for bounded variation functions

any  $v \in BV(\Omega)$ . For  $1 \leq p < \frac{d}{d-1}$  this embedding is compact; cf. [2]. Smooth functions are dense in  $BV(\Omega) \cap L^p(\Omega)$ ,  $1 \leq p < \infty$ . In particular, for  $v \in BV(\Omega) \cap L^2(\Omega)$  and  $\delta > 0$  there exists  $\varepsilon := \varepsilon(\delta) > 0$  and functions  $(v_\varepsilon)_{\varepsilon>0} \subset C^\infty \cap BV(\Omega) \cap L^2(\Omega)$  such that

$$\|\nabla v_\varepsilon\|_{L^1(\Omega)} \leq |Dv|(\Omega) + c_0\delta, \quad (13.3)$$

$$\|v - v_\varepsilon\|_{L^2(\Omega)} \leq c_1\delta, \quad \|v - v_\varepsilon\|_{L^1(\Omega)} \leq c_2\delta, \quad (13.4)$$

$$\|D^2v_\varepsilon\|_{L^2(\Omega)} \leq \varepsilon^{-2}\|v\|_{L^2(\Omega)}, \quad \|D^2v_\varepsilon\|_{L^1(\Omega)} \leq \varepsilon^{-2}\|v\|_{L^1(\Omega)}, \quad (13.5)$$

cf. [3]. These inequalities follow from standard mollifier techniques, see e.g. [13]. Now we are able to show the following convergence result, which follows similar ideas as the proof of [3, Theorem 3.1].

**Theorem 13.1.** *Let  $u_h \in \arg \min_{v \in \mathcal{S}^1(\mathcal{T}_h)} J_{\alpha_1, \alpha_2}(v)$  and  $u \in BV(\Omega) \cap L^2(\Omega)$  be a minimizer of the function  $J_{\alpha_1, \alpha_2}$ . Then we have that  $J_{\alpha_1, \alpha_2}(u_h) \rightarrow J_{\alpha_1, \alpha_2}(u)$  as  $h \rightarrow 0$ . If additionally  $\alpha_2 > 0$ , then  $u_h \rightarrow u$  in  $L^2(\Omega)$  as  $h \rightarrow 0$ .*

*Proof.* By the optimality of  $u$  we have  $J_{\alpha_1, \alpha_2}(u_h) - J_{\alpha_1, \alpha_2}(u) \geq 0$ . For  $\delta > 0$  let  $u_\varepsilon \in C^\infty(\Omega) \cap L^2(\Omega)$  as above and  $\mathcal{I}_h u_\varepsilon$  its nodal interpolant, i.e.,  $\mathcal{I}_h u_\varepsilon \in \mathcal{S}^1(\mathcal{T}_h)$ . Then we deduce

$$\begin{aligned} J_{\alpha_1, \alpha_2}(u_h) - J_{\alpha_1, \alpha_2}(u) &\leq J_{\alpha_1, \alpha_2}(\mathcal{I}_h u_\varepsilon) - J_{\alpha_1, \alpha_2}(u) \\ &= \|\nabla \mathcal{I}_h u_\varepsilon\|_{L^1(\Omega)} + \alpha_1 \|\mathcal{I}_h u_\varepsilon - g\|_{L^1(\Omega)} + \alpha_2 \|\mathcal{I}_h u_\varepsilon - g\|_{L^2(\Omega)}^2 \\ &\quad - |Du|(\Omega) - \alpha_1 \|u - g\|_{L^1(\Omega)} - \alpha_2 \|u - g\|_{L^2(\Omega)}^2. \end{aligned}$$

Using (13.3) and  $\|\mathcal{I}_h u_\varepsilon - g\|_{L^2(\Omega)}^2 - \|u - g\|_{L^2(\Omega)}^2 = \int_\Omega (\mathcal{I}_h u_\varepsilon - u)(\mathcal{I}_h u_\varepsilon + u - 2g)$  we obtain

$$\begin{aligned} J_{\alpha_1, \alpha_2}(u_h) - J_{\alpha_1, \alpha_2}(u) &\leq \|\nabla \mathcal{I}_h u_\varepsilon\|_{L^1(\Omega)} - \|\nabla u_\varepsilon\|_{L^1(\Omega)} + c_0\delta + \alpha_1 \|\mathcal{I}_h u_\varepsilon - g\|_{L^1(\Omega)} \\ &\quad - \alpha_1 \|u - g\|_{L^1(\Omega)} + \alpha_2 \int_\Omega (\mathcal{I}_h u_\varepsilon - u)(\mathcal{I}_h u_\varepsilon + u - 2g). \end{aligned}$$

By the triangle-inequality and the Cauchy-Schwarz inequality we get

$$\begin{aligned} J_{\alpha_1, \alpha_2}(u_h) - J_{\alpha_1, \alpha_2}(u) &\leq \|\nabla(\mathcal{I}_h u_\varepsilon - u)\|_{L^1(\Omega)} + c_0\delta + \alpha_1 \|\mathcal{I}_h u_\varepsilon - g - (u - g)\|_{L^1(\Omega)} \\ &\quad + \alpha_2 \|\mathcal{I}_h u_\varepsilon - u\|_{L^2(\Omega)} (\|\mathcal{I}_h u_\varepsilon\|_{L^2(\Omega)} + \|u\|_{L^2(\Omega)} + 2\|g\|_{L^2(\Omega)}) \\ &= \|\nabla(\mathcal{I}_h u_\varepsilon - u_\varepsilon)\|_{L^1(\Omega)} + c_0\delta + \alpha_1 \|\mathcal{I}_h u_\varepsilon - u_\varepsilon + u_\varepsilon - u\|_{L^1(\Omega)} \\ &\quad + \alpha_2 \|\mathcal{I}_h u_\varepsilon - u_\varepsilon + u_\varepsilon - u\|_{L^2(\Omega)} (\|\mathcal{I}_h u_\varepsilon\|_{L^2(\Omega)} + \|u\|_{L^2(\Omega)} + 2\|g\|_{L^2(\Omega)}) \\ &\leq \|\nabla(\mathcal{I}_h u_\varepsilon - u_\varepsilon)\|_{L^1(\Omega)} + c_0\delta + \alpha_1 \|\mathcal{I}_h u_\varepsilon - u_\varepsilon\|_{L^1(\Omega)} + \alpha_1 \|u_\varepsilon - u\|_{L^1(\Omega)} \\ &\quad + \alpha_2 (\|\mathcal{I}_h u_\varepsilon - u_\varepsilon\|_{L^2(\Omega)} + \|u_\varepsilon - u\|_{L^2(\Omega)}) (\|\mathcal{I}_h u_\varepsilon\|_{L^2(\Omega)} + \|u\|_{L^2(\Omega)} + 2\|g\|_{L^2(\Omega)}). \end{aligned}$$

The bound  $\|\mathcal{I}_h u_\varepsilon\|_{L^2(\Omega)} + \|u\|_{L^2(\Omega)} + 2\|g\|_{L^2(\Omega)} \leq \tilde{c}$ , which holds provided that  $h \leq \varepsilon$ , and the nodal interpolant estimate yield

$$\begin{aligned} J_{\alpha_1, \alpha_2}(u_h) - J_{\alpha_1, \alpha_2}(u) &\leq c_I h \|D^2 u_\varepsilon\|_{L^1(\Omega)} + c_0\delta + c_I \alpha_1 h^2 \|D^2 u_\varepsilon\|_{L^1(\Omega)} + \alpha_1 \|u_\varepsilon - u\|_{L^1(\Omega)} \\ &\quad + \tilde{c} \alpha_2 (c_I h^2 \|D^2 u_\varepsilon\|_{L^2(\Omega)} + \|u_\varepsilon - u\|_{L^2(\Omega)}). \end{aligned}$$

Using (13.4), (13.5), and the bound  $\|u\|_{L^2(\Omega)} \leq \tilde{c}$  we get

$$J_{\alpha_1, \alpha_2}(u_h) - J_{\alpha_1, \alpha_2}(u) \leq C_1 \frac{h}{\varepsilon^2} + C_2 \delta + C_3 \frac{h^2}{\varepsilon^2} + C_4 \delta + C_5 \frac{h^2}{\varepsilon^2} + C_6 \delta.$$

### III. Finite Element Discretization of the Total Variation

Let  $h$  be sufficiently small, i.e.,  $h \leq \min\{\delta, \delta\varepsilon^2\}$ . Then for  $\delta \rightarrow 0$  we deduce that  $h \rightarrow 0$  and hence  $J_{\alpha_1, \alpha_2}(u_h) \rightarrow J_{\alpha_1, \alpha_2}(u)$ .

If  $\alpha_2 > 0$ , then  $J_{\alpha_1, \alpha_2}$  is strictly convex and it follows that

$$J_{\alpha_1, \alpha_2}(u_h) - J_{\alpha_1, \alpha_2}(u) \geq \alpha_2 \|u_h - u\|_{L^2(\Omega)}^2,$$

cf. [19, Lemma 3.8]. Hence  $u_h \rightarrow u$  for  $h \rightarrow 0$ .  $\square$

By the definition of the total variation (13.2) an equivalent formulation of (13.1) reads

$$\min_{v \in BV(\Omega) \cap L^2(\Omega)} \max_{\mathbf{p} \in C_0^\infty(\Omega, \mathbb{R}^d), |\mathbf{p}|_2 \leq 1} \alpha_1 \|v - g\|_{L^1(\Omega)} + \alpha_2 \|v - g\|_{L^2(\Omega)}^2 + \int_\Omega v \operatorname{div} \mathbf{p}. \quad (13.6)$$

An analog equivalence exists for finite element spaces [3]. In particular, for  $v \in \mathcal{S}^1(\mathcal{T}_h)$  we have that

$$|Dv|(\Omega) = \sup_{\mathbf{p} \in \mathcal{L}_N^0(\mathcal{T}_h)^d, |\mathbf{p}|_2 \leq 1} \int_\Omega \nabla v \cdot \mathbf{p} \, dx \quad (13.7)$$

leading to

$$\inf_{v \in \mathcal{S}^1(\mathcal{T}_h)} J_{\alpha_1, \alpha_2}(v) = \inf_{v \in \mathcal{S}^1(\mathcal{T}_h)} \sup_{\mathbf{p} \in \mathcal{L}_N^0(\mathcal{T}_h)^d} \int_\Omega \nabla v \cdot \mathbf{p} \, dx + \alpha_1 \|v - g\|_{L^1(\Omega)} + \alpha_2 \|v - g\|_{L^2(\Omega)}^2 - I_K(\mathbf{p}),$$

where

$$I_K(\mathbf{p}) := \begin{cases} 0 & \text{if } \mathbf{p} \in \mathcal{K} \\ \infty & \text{else} \end{cases}$$

is the indicator function of  $\mathcal{K} := \{\mathbf{p} \in L^1(\Omega)^d : |\mathbf{p}|_2 \leq 1\}$ . For  $v \in \mathcal{L}^0(\mathcal{T}_h)$  we obtain

$$|Dv|(\Omega) = \sup_{(\beta_S)_{S \in \mathcal{S}_h}, |\beta_S|_2 \leq 1} \sum_{S \in \mathcal{S}_h} |S| \beta_S[v]_S, \quad (13.8)$$

where  $[v]_S \in \mathbb{R}$  defines the jump across  $S \in \mathcal{S}_h$  in normal direction. Hence in this case the discrete problem reads as

$$\inf_{v \in \mathcal{L}^0(\mathcal{T}_h)} J_{\alpha_1, \alpha_2}(v) = \inf_{v \in \mathcal{L}^0(\mathcal{T}_h)} \sup_{\mathbf{p} \in \mathcal{L}_N^0(\mathcal{S}_h)} \sum_{S \in \mathcal{S}_h} |S| \mathbf{p}|_S[v]_S + \alpha_1 \|v - g\|_{L^1(\Omega)} + \alpha_2 \|v - g\|_{L^2(\Omega)}^2 - I_K(\mathbf{p}).$$

While this formulation is equivalent to the primal formulation, in general we cannot expect convergence to the continuous solution, unless all the jumps are correctly represented by a descendant triangulation. For more details we refer the reader to [3, Section 4].

The primal-dual problem does not have a unique solution in general, even if the primal problem is strictly convex ( $\alpha_2 > 0$ ). This is due to the fact, that the dual problem of (13.1), even for  $\alpha_1 = 0$  and  $\alpha_2 > 0$ , is only convex but not strictly convex, see for example [15, 18].

#### 13.2.2. Primal-Dual Algorithm

In the following we discretize using finite element spaces  $\mathcal{U}$  (e.g.  $\mathcal{S}^1(\mathcal{T}_h)$  or  $\mathcal{L}^0(\mathcal{T}_h)$ ) and  $\mathcal{P}$  (e.g.  $\mathcal{S}_0^1(\mathcal{T}_h)^d$  or  $\mathcal{L}_N^0(\mathcal{T}_h)^d$ ) for the primal variable  $u$  and the dual variable  $\mathbf{p}$ , respectively. We denote by  $\langle \cdot, \cdot \rangle$  the  $L^2$ -inner product or the application of an  $L^2$ -functional. Following the

### 13. Using DUNE-ACFem for bounded variation functions

ideas of Chambolle and Pock [10, Algorithm 1] we formulate our primal-dual algorithm by identifying  $F^* : \mathcal{P} \rightarrow \mathbb{R} \cup \{+\infty\}$ ,  $G : \mathcal{U} \rightarrow \mathbb{R} \cup \{+\infty\}$ , and  $K : \mathcal{U} \rightarrow \mathcal{P}^*$  with

$$F^*(\mathbf{p}) = I_{\mathcal{K}}(\mathbf{p}), \quad G(v) = \alpha_1 \|v - g\|_{L^1(\Omega)} + \alpha_2 \|v - g\|_{L^2(\Omega)}^2, \quad \text{and } \langle Kv, \mathbf{p} \rangle = \int_{\Omega} v \operatorname{div} \mathbf{p},$$

where we assume that each element in  $\mathcal{P}$  has a weak derivative. If this is not the case,  $\langle Kv, \mathbf{p} \rangle$  is to be understood using the identifications suggested by the equations (13.7) and (13.8). That is  $\langle Kv, \mathbf{p} \rangle = \int_{\Omega} \nabla v \cdot \mathbf{p} \, dx$  and

$$\langle Kv, \mathbf{p} \rangle = \int_{\Omega} v \operatorname{div} \mathbf{p} = \sum_{T \in \mathcal{T}_h} \int_{\partial T} v \mathbf{p} \cdot \mathbf{n} = \sum_{S \in \mathcal{S}_h} [v]_S \mathbf{n} \cdot \int_S \mathbf{p} = \sum_{S \in \mathcal{S}_h} |S| \mathbf{p}|_S [v]_S,$$

cf. [3, Lemma 4.1], respectively.

In the case that  $\nabla v \notin L^2(\Omega)$  for  $v \in \mathcal{U}$  or  $\operatorname{div} \mathbf{q} \notin L^2(\Omega)$  for  $\mathbf{q} \in \mathcal{P}$  we use the following identities

$$\begin{aligned} \mathcal{U} &= \mathcal{L}^0(\mathcal{T}_h), \mathcal{P} = \mathcal{L}_N^0(\mathcal{S}_h) : & \langle \nabla v, \mathbf{q} \rangle &:= - \sum_{S \in \mathcal{S}_h} |S| \mathbf{p}|_S [v]_S =: -\langle v, \operatorname{div} \mathbf{q} \rangle, \\ \mathcal{U} &= \mathcal{L}^0(\mathcal{T}_h), \mathcal{P} = \mathcal{S}_0^1(\mathcal{T}_h) : & \langle \nabla v, \mathbf{q} \rangle &:= -\langle v, \operatorname{div} \mathbf{q} \rangle, \\ \mathcal{U} &= \mathcal{S}^1(\mathcal{T}_h), \mathcal{P} = \mathcal{L}_N^0(\mathcal{T}_h) : & \langle \nabla v, \mathbf{q} \rangle &=: -\langle v, \operatorname{div} \mathbf{q} \rangle. \end{aligned} \tag{13.9}$$

We assume, that the datum  $g \in L^2(\Omega)$  and the cost parameters  $\alpha_i$  are given. Then in our algorithm we initialize  $u_0 = \bar{u}_0 \in \mathcal{U}$ , e.g.,  $u_0 \equiv 0$  or  $u_0 = \mathbb{P}g$ , where  $\mathbb{P} : L^2(\Omega) \rightarrow \mathcal{U}$  denotes the  $L^2$ -projection onto the discrete space  $\mathcal{U}$ , and  $\mathbf{p}_0 \in \mathcal{P}$  to be the zero function. As parameters we need the step sizes  $\sigma, \tau > 0$ , the coefficient  $\beta = \frac{\tau \alpha_1}{1 + 2\tau \alpha_2}$ , and the overrelaxation parameter  $\theta \in [0, 1]$ .

Then our primal-dual algorithm iterates starting with  $k = 0$  as follows:

1. Set  $\bar{\mathbf{p}} \in \mathcal{P}$  with step size  $\sigma$  as

$$\langle \bar{\mathbf{p}}, \mathbf{q} \rangle = \langle \mathbf{p}_k + \sigma \nabla \bar{u}_k, \mathbf{q} \rangle \quad \forall \mathbf{q} \in \mathcal{P}. \tag{13.10}$$

If  $\nabla \bar{u}_k \in \mathcal{P}$ , then we can use the strong formulation, otherwise we use the identities (13.9). As the algorithm is derived from formulation (13.6), we need to guarantee that  $|\mathbf{p}_k|_2 \leq 1$  for all  $k$ . This is done by the following update

$$\mathbf{p}_{k+1} = (I + \sigma \partial F^*)^{-1}(\bar{\mathbf{p}}) \quad \Leftrightarrow \quad \mathbf{p}_{k+1}(x) = \frac{\bar{\mathbf{p}}(x)}{\max(|\bar{\mathbf{p}}(x)|_2, 1)}, \tag{13.11}$$

for almost any  $x \in \Omega$ . Note, that due to the structure of the operator  $(I + \sigma \partial F^*)^{-1}$  (see [10] for more details)  $\mathbf{p}_{k+1} \in \mathcal{P} \cap \mathcal{K}$ .

2. Update  $u_k$  using

$$u_{k+1} = (I + \tau \partial G)^{-1}(u_k + \tau \operatorname{div} \mathbf{p}_{k+1}) \quad \Leftrightarrow \quad u_{k+1}(x) = \begin{cases} z(x) - \beta & \text{if } z(x) - \beta \geq \mathbb{P}g(x) \\ z(x) + \beta & \text{if } z(x) + \beta \leq \mathbb{P}g(x) \\ \mathbb{P}g(x) & \text{else ,} \end{cases} \tag{13.12}$$

### III. Finite Element Discretization of the Total Variation

for almost any  $x \in \Omega$ , where  $z \in \mathcal{U}$  is defined via

$$\langle z, v \rangle = \frac{1}{1 + 2\tau\alpha_2} (\langle u_k + 2\tau\alpha_2 g, v \rangle + \tau \langle \operatorname{div} \mathbf{p}_{k+1}, v \rangle) \quad \forall v \in \mathcal{U}. \quad (13.13)$$

If the  $\operatorname{div} \mathbf{p}_{k+1} \notin L^2(\Omega)$ , then we use again the identities of (13.9).

3. As proposed in [10] we overrelaxate to speed up the algorithm:

$$\bar{u}_{k+1} = u_{k+1} + \theta(u_{k+1} - u_k) \quad (13.14)$$

4. If the stopping criterion

$$\frac{\|u_{k+1} - u_k\|_{L^2}}{\tau} + \frac{\|\mathbf{p}_{k+1} - \mathbf{p}_k\|_{L^2}}{\sigma} < TOL \quad (13.15)$$

holds, we terminate the algorithm, otherwise we set  $k \rightarrow k + 1$  and repeat.

Note that in equation (13.11) with  $\mathcal{P} = \mathcal{S}_0^1(\mathcal{T}_h)^d$  it is a priori not clear that  $\mathbf{p}_{k+1} \in \mathcal{S}_0^1(\mathcal{T}_h)^d$ , as taking the pointwise maximum of two piecewise linear functions results in a piecewise linear function on a finer grid. So in this case we additionally apply the nodal interpolation operator  $\mathcal{I}_h : C^0(\Omega)^d \rightarrow \mathcal{P}$ , which then guarantees  $\|\mathbf{p}_{k+1}\|_{L^\infty} \leq 1$  and  $\mathbf{p}_{k+1} \in \mathcal{P}$ . A similar problem arises for  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$  and equation (13.12), which we treat analogously. Furthermore note, that testing with the complete test-space  $\mathcal{U}$  or  $\mathcal{P}$  (eqs. (13.10), (13.13)), respectively, is equivalent to an  $L^2$ -projection onto the respective space.

Theorem 1 of [10] guarantees convergence of this algorithm to a saddle point, if the according discretization of problem (13.6) has a saddle point and additionally  $\theta = 1$  and

$$\sigma\tau B^2 < 1, \quad (13.16)$$

where  $B := \|\nabla\|_{L^2}$  is the operator norm of the gradient operator on the discrete space  $\mathcal{U}$ . This norm is bounded for discrete functions and of order  $O(1/h_{\min})$ , where  $h_{\min} := \min_{T \in \mathcal{T}_h} \operatorname{diam}(T)$ , which implies that using finer meshes will always result in more steps of the primal-dual algorithm. Even using an adaptive mesh is not beneficial to the stepsize, as  $B$  depends on the minimum mesh-width.

### 13.3. Implementation Details

The algorithm is implemented in the DUNE-project `non-smooth-minimization`. This project is compatible with the 2.4-release and available on the website <http://www.ians.uni-stuttgart.de/nmh/downloads>. It requires the modules DUNE-ACFEM and all modules that are required by it, available at [gitlab.dune-project.org](https://gitlab.dune-project.org). The installation works the standard DUNE-way. (see Section 13.6)

#### 13.3.1. On DUNE-ACFEM

DUNE-ACFEM [14] is a simulation framework based on DUNE-FEM [11], which is a discretization framework based on DUNE. DUNE-FEM provides most generally speaking finite-element spaces on generic grids and all the corresponding utilities to construct a finite-element scheme. Examples are given in the DUNE-FEM-HOWTO. For more information consult [11].

### 13. Using DUNE-ACFem for bounded variation functions

The add-on module DUNE-ACFEM provides expression templates (and also analytical functions, that can be evaluated on the mesh) for the discrete functions of DUNE-FEM and also for models similar but more elaborate to those in the DUNE-FEM-HOWTO. This aims at simplifying the algorithmic formulation of elliptic and parabolic PDEs. The expression templates for discrete functions allow for addition, multiplication with scalars, multiplication with scalar functions, scalar products of the components of two functions, and unary expressions like componentwise sin, cos, sqrt, exp.

DUNE-ACFEM is designed to treat 2nd order elliptic PDEs of the form

$$\begin{aligned} -\nabla \cdot (A(x, u, \nabla u) \nabla u) + \nabla \cdot (b(x, u, \nabla u) u) + c(x, u, \nabla u) u &= f(x) \quad \text{in } \Omega, \\ u &= g_D \quad \text{on } \Gamma_D, \\ (A(x, u, \nabla u) \nabla u) \cdot \nu + \alpha(x, u) u &= g_N \quad \text{on } \Gamma_R, \\ (A(x, u, \nabla u) \nabla u) \cdot \nu &= g_N \quad \text{on } \Gamma_N, \end{aligned} \tag{13.17}$$

or, in weak formulation,

$$\begin{aligned} \int_{\Omega} (A \nabla u) \cdot \nabla \phi dx + \int_{\Omega} (\nabla \cdot (b u) + c u) \phi dx - \int_{\Omega} f(x) \phi dx \\ + \int_{\Gamma_R} \alpha u \phi do - \int_{\Gamma_N \cup \Gamma_R} g_N \phi do = 0, \\ \langle \Pi, u \rangle = \langle \Pi, g_D \rangle. \end{aligned} \tag{13.18}$$

Possible Dirichlet data is enforced by standard Lagrange test-functions  $\Pi$ . The multiplication with the test-functions  $\phi$  and the integral (quadrature) is provided by DUNE-FEM. The other part of the integral has to be provided by the model, which is the central concept of DUNE-ACFEM. A model is basically a tuple of the form

$$\mathcal{M} := (\sigma_E, \mu_E, \rho_I, g_N, g_D, w_D, f, \chi_R, \chi_N, \chi_D, \Psi)$$

with the following constituents:

Flux	$\sigma_E : \mathbb{R}^d \times \mathbb{R}^m \times \mathbb{R}^{m \times d} \rightarrow \mathbb{R}^{m \times d}$ ,
Source	$\mu_E : \mathbb{R}^d \times \mathbb{R}^m \times \mathbb{R}^{m \times d} \rightarrow \mathbb{R}^m$ ,
Robin-flux	$\rho_I : \mathbb{R}^d \times \mathbb{R}^m \rightarrow \mathbb{R}^m$ ,
Robin-indicator	$\chi_R : \partial\Omega \rightarrow \{0, 1\}$ ,
Neumann-data	$g_N : \mathbb{R}^d \rightarrow \mathbb{R}^m$ ,
Neumann-indicator	$\chi_N : \partial\Omega \rightarrow \{0, 1\}$ ,
Dirichlet-data	$g_D : \mathbb{R}^d \rightarrow \mathbb{R}^m$ ,
Dirichlet-indicator	$\chi_D : \partial\Omega \rightarrow \{0, 1\}$ ,
Bulk-forces	$f : \mathbb{R}^d \rightarrow \mathbb{R}^m$ ,
Force-functional	$\Psi \in \mathcal{V}^*$ .

Using such a model results in the following weak formulation

$$\begin{aligned} \sum_{E \in \mathcal{T}} \left( \int_E (\sigma_E(x, U, \nabla U) : \nabla V + \mu_E(x, U, \nabla U) \cdot V - f \cdot V) \right. \\ \left. + \int_{I \in E \cap \partial\Omega} ((\chi_R(x) \rho_I(x, U) - \chi_N(x) g_N(x)) \cdot V) \right) - \langle \Psi, V \rangle = 0, \\ \langle \Pi, \chi_D U \rangle = \langle \Pi, \chi_D g_D \rangle. \end{aligned} \tag{13.19}$$

This is directly reflected in code, as a model is derived from an interface class that requires exactly the implementation of the above constituents (and the linearization of Flux, Source

### III. Finite Element Discretization of the Total Variation

and Robin-flux for nonlinear models). Each model is derived from a default *zero-model*, where all the constituents are set to zero, so only nonzero contributions need to be implemented. DUNE-ACFEM allows forming algebraic expressions from existing models. The generated model-expressions fulfill the model-interface just as "hand-coded" ones. This allows forming of complicated models from a set of convenient pre-built skeleton-models. For a list of predefined models consult [14]. The algebraic expressions include linear combinations of models and multiplying with  $L^\infty$ -functions.

Other important concepts are the `FemScheme` and the `EllipticOperator`. The `EllipticOperator` applies or assembles the discretization of equation (13.18). The `FemScheme` then solves the given problem and chooses the necessary linear or nonlinear solvers, depending on structural information given by the model. DUNE-ACFEM mostly uses preconditioned iterative solvers like CG or GMRes.

#### 13.3.2. Image Read-In and Noise Simulation

To read-in images we use the CIMG Library - C++ Template Image Processing Toolkit [21]. The single header file library CIMG is an open source project to provide easy handling and processing of images. It is capable of reading-in standard image formats (e.g. `.png`, `.tif`, `.jpg`) and of adding certain types of noise. Each pixel is accessible by its coordinates utilizing the data type `image`, whose constructor gets the filename containing the image location. Moreover, `image` provides a method to add noise to the data with a given noise level for different noise-types. In particular, Gaussian noise with noise level (variance)  $\gamma$  and mean 0 and salt-and-pepper noise  $S$  with noise level  $s \in [0, 1]$ , which corrupts an original image  $\hat{g}$  by

$$S\hat{g}(x) = \begin{cases} \min \hat{g} & \text{with probability } s/2 \\ \max \hat{g} & \text{with probability } s/2 \\ \hat{g}(x) & \text{with probability } 1 - s \end{cases},$$

are available.

We define two adapter classes within the header file `src/datafunction.hh` to convert either the data type `image` or an algebraic expression into a DUNE-FEM discrete function. They derive from the `GridFunctionAdapter` that creates discrete functions from an evaluation method on a mesh element. These functions are defined on the unit square  $[0, 1] \times [0, 1]$ , as we can scale any rectangular image accordingly. The image resolution and aspect ratio relate to the level of initial refinement and to the number of cells in each direction. So  $n$  uniform refinements with  $l \times k$  initial cells lead to an image resolution of size  $l2^n \times k2^n$  square pixels, where every square is composed of two triangles and  $l/k$  corresponds to the aspect ratio. This is done in the grid-file `data/unitsquare2d.dgf` which describes the unit square using the DUNE DGF – Interval [6] nomenclature.

The artificial addition of noise (`NSM_USE_NOISE=true`) is only needed for demonstration purposes as in real-world application the image is already corrupted by noise due to certain physical processes. We currently add two independent types of noise, where every noise implemented by CIMG can be applied. Noise type and noise level are controlled by a set of parameters defined in the parameter file. For instance, setting `nsm.noiseType1` to 0 leads to Gaussian noise and setting it to 2 corresponds to salt-and-pepper noise. Setting any noise level parameter to 0 disables the respective noise. For other noise types and more information consult the documentation of CIMG .

### 13.3.3. Implementation of the Primal-Dual Algorithm

The primal-dual algorithm from Section 13.2.2 is implemented in the file `src/nsm.cc`, where some of its subroutines, which depend on the continuity of the discrete spaces  $\mathcal{U}$  and  $\mathcal{P}$ , are outsourced into the file `src/phc.hh`. The template class `ProjectionHelperClass` uses partial template specialization to correctly define the methods `calculateZ()` and `entitywiseProjection()`.

The method `phc.entitywiseProjection()` realizes equations (13.10) and (13.11). Solving eq. (13.10) is relatively easy, as all steps are provided by DUNE-ACFEM, e.g. for continuous  $\mathcal{U}$  and  $\mathcal{P}$ :

```
void entitywiseProjection ( const ForwardDiscreteFunctionType & uBar ,
                            AdjointDiscreteFunctionType & p )
{
    //the gradient model defines the weak gradient using continuous test-functions
     //from the space of p
    auto gradU = gradientModel(uBar);
    auto Dbc0 = dirichletZeroModel(p);
    auto mass = massModel(p);
    //we solve p = sigma * nabla u + p
     //with dirichlet zero boundary
    auto model = mass - sigma_ * gradU - p + Dbc0 ;
    //Testspace = AdjointDiscreteFunctionSpaceType = space of p
    typedef EllipticFemScheme<AdjointDiscreteFunctionType , decltype(model)> SchemeType
    ;
    //p is the returned solution
    SchemeType scheme(p, model);
    scheme.solve();
}
```

In the above case, for given  $\mathbf{p}$  and  $\bar{u}$  the equation

$$0 = \langle \bar{p}, \phi \rangle - \sigma \langle \bar{u}, -\operatorname{div} \phi \rangle - \langle \mathbf{p}, \phi \rangle \quad \forall \phi \in \mathcal{P}$$

with Dirichlet-zero boundary conditions is solved with respect to  $\bar{p}$ . The solution is written into the variable  $\mathbf{p}$ .

Projecting  $\mathbf{p}$  to be feasible (eq. (13.11)) is not in the features of DUNE-ACFEM. So we have to use the features of DUNE-FEM directly. We choose to do the projection entity-wise, i.e. we iterate over all entities of the grid. On each entity we iterate over the degrees of freedom and if  $|\mathbf{p}|_2 > 1$  holds, we restrict the corresponding value to length 1 in the same direction. If the polynomial order of  $\mathcal{P}$  is less or equal than 1, this implies the necessary condition  $\|\mathbf{p}\|_{L^\infty} \leq 1$ . For the space  $\mathcal{S}^1(\mathcal{T}_h)$  this implies the application of the nodal interpolant  $\mathcal{I}_h$ .

The calculation of  $z$  (eq. (13.13)) translates very nicely into code. We use the `L2Projection` of DUNE-ACFEM and its ability to linearly combine discrete functions. If  $\mathcal{P}$  is continuous, the weak form of eq. (13.13) is

$$\langle z_k, \psi \rangle = \frac{1}{1 + 2\tau\alpha_2} \langle u_k + \tau \operatorname{div} \mathbf{p}_{k+1} + 2\tau\alpha_2 g, \psi \rangle \quad \forall \psi \in \mathcal{U}$$

and translates into code in the following way:

```
void calculateZ ( const AdjointDiscreteFunctionType &p , const
                    ForwardDiscreteFunctionType & uOld , ForwardDiscreteFunctionType& z)
{
    auto divP = divergence(p);
    L2Projection(1. / (1. + tau_*lambda_2_) * ( tau_ * divP + uOld + tau_ * lambda_2_ *
        projG_), z);
}
```

### III. Finite Element Discretization of the Total Variation

For  $\mathcal{P} = \mathcal{L}_N^0(\mathcal{T}_h)^d$  we use the weak divergence to shift the derivative to the test-space  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$ . As above we use the basic Models of DUNE-ACFEM, in particular the `massModel` and the `weakDivergenceModel` to implement eq. (13.13) as follows:

```
void calculateZ (const AdjointDiscreteFunctionType &p, const
    ForwardDiscreteFunctionType & uOld, ForwardDiscreteFunctionType& z)
{
    //get the mass model of the Forward space
    auto U_Phi = massModel(projG_.space());
    //calculate z
    auto weakDiv_P = weakDivergenceModel(p);
    auto projModel = U_Phi -1./(1.+tau_*lambda_2_) * ( tau_* weakDiv_P + uOld + tau_
        * lambda_2_ * projG_);
    typedef EllipticFemScheme<ForwardDiscreteFunctionType, decltype(projModel)>
        SchemeType;
    SchemeType scheme(z, projModel);
    scheme.solve();
}
```

Note that in contrast to the implementation of the method `entitywiseProjection()` we construct the `massModel` from a discrete space instead of a discrete function. This is explicitly allowed by DUNE-ACFEM as the object simply has to provide the data type for the test-function space.

For the case of  $\mathcal{P} = \mathcal{L}_N^0(\mathcal{S}_h)^d$  and  $\mathcal{U} = \mathcal{L}^0(\mathcal{T}_h)$  the summand `weakDiv_p` is turned into a functional. This functional is called `EdgeWeakDivergenceFunctional` and is contained in the file `edgeweakdivergencefunctional.hh`. The main implementation is done in the `coefficients()` method, that implements the application of the functional to the basis functions in DUNE-FEM notation.

In the resulting code for the method `calculateZ()`, (see code example above) the main difference is that `weakDivergenceModel(p)` is replaced by `edgeWeakDivergenceFunctional(p)`. Additionally `weakDiv_p` in the definition of `projModel` has been moved outside the brackets, as functions are not allowed to be added to functionals outside a model (see below).

```
auto weakDiv_P = edgeWeakDivergenceFunctional(p);
auto projModel = U_Phi -1./(1.+tau_*lambda_2_) * ( uOld + tau_* lambda_2_ *
    projG_) - tau_/(1.+tau_*lambda_2_) * weakDiv_p;
```

#### 13.3.4. Adaptive Refinement

Adaptive refinement is initiated by a positive parameter `nsm.localRefine`. This parameter denotes the number of additional local refinements to be done in the initialization phase in addition to the uniform refinements. If it is set to  $\leq 0$ , no local refinement will be performed. The grid is locally refined at discontinuities of the piecewise constant datum  $g \in \mathcal{L}^0(\mathcal{T}_h)$  in the following way: Given an entity  $E$  and its neighbour  $N$ , if

$$|g_{|N} - g_{|E}| > ADAPTTOL \quad (13.20)$$

holds, then  $E$  is marked for refinement, where the value of `ADAPTTOL` is given by the parameter `nsm.adaptTolerance`. We do not refine during the iterations of the primal-dual algorithm, but keep the mesh static. So the refinement increases the resolution of discontinuities and hence drastically improves the projection  $\mathbb{P}g$ . This is implemented in the method `adaptGrid()` in the file `nsm.cc`. We use the grid manager DUNE-ALUGRID [1], which is capable of handling the needed conforming, parallel, adaptive, triangular grids.

## 13.4. Numerical Experiments

To reproduce the data of the experiments of this section, consult Section 13.6 and follow the described steps.

### 13.4.1. Comparison of Discrete Spaces

We investigate for different discrete spaces  $\mathcal{U}$  and  $\mathcal{P}$  the behaviour of our algorithm with respect to the nonsmooth minimization problem (13.1) considering the following setups

$$\begin{aligned} \mathcal{U} &= \mathcal{S}^1(\mathcal{T}_h), \mathcal{P} = \mathcal{S}_0^1(\mathcal{T}_h)^d, & \mathcal{U} &= \mathcal{L}^0(\mathcal{T}_h), \mathcal{P} = \mathcal{S}_0^1(\mathcal{T}_h)^d, \\ \mathcal{U} &= \mathcal{S}^1(\mathcal{T}_h), \mathcal{P} = \mathcal{L}_N^0(\mathcal{T}_h)^d, & \mathcal{U} &= \mathcal{L}^0(\mathcal{T}_h), \mathcal{P} = \mathcal{L}_N^0(\mathcal{S}_h)^d. \end{aligned}$$

The setups using  $\mathcal{P} = \mathcal{L}_N^0(\dots)^d$  are motivated by the equivalence of the mixed formulation and the primal formulation for these discrete settings. For the case  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h), \mathcal{P} = \mathcal{L}_N^0(\mathcal{T}_h)^d$  we even have guaranteed convergence to the continuous formulation, see Theorem 13.1. The setting  $\mathcal{P} = \mathcal{S}_0^1(\mathcal{T}_h)^d$  is motivated by [5], where it is shown that for  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h), \mathcal{P} = \mathcal{S}_0^1(\mathcal{T}_h)^d$  the discrete pre-dual of the functional  $J_{0,\alpha_2}$   $\Gamma$ -converges to the continuous one. However, setting  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$  does not seem necessary for the proof there, but can be for example replaced by choosing  $\mathcal{U} = \mathcal{L}^0(\mathcal{T}_h)$ . Therefore we also investigate the case  $\mathcal{U} = \mathcal{L}^0(\mathcal{T}_h)$ .

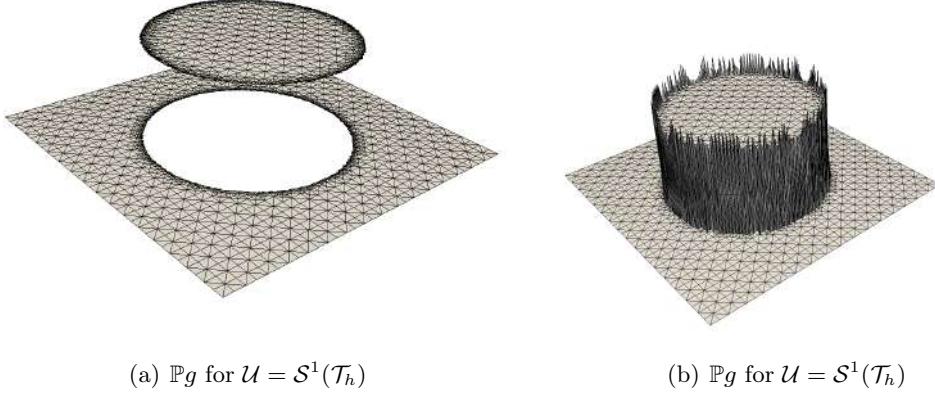
For this set of experiments we choose a similar example as in [5], i.e., the observed data is 1 on a disk of radius 0.3 and 0 elsewhere. Hence, we choose the discrete datum  $g$  to be given as a piecewise constant approximation of the function

$$f(x) = \begin{cases} 1, & \text{if } |x - (0.5, 0.5)| \leq 0.3 \\ 0, & \text{else.} \end{cases}$$

Note that this datum changes under grid refinement. The cost parameters are set to  $\alpha_1 = 10$  and  $\alpha_2 = 20$ , the stopping tolerance to  $10^{-2}$ , and the adaptation tolerance to 0.1. For the  $\mathcal{L}^0/\mathcal{L}^0$  case we choose the tolerance  $10^{-4}$  as we use an extension of  $\mathcal{P}$  into  $\Omega$  to calculate the part of  $\mathbf{p}$  in the stopping criterion  $\frac{\|u_{k+1} - u_k\|_{L^2}}{\tau} + \frac{\|\mathbf{p}_{k+1} - \mathbf{p}_k\|_{L^2}}{\sigma} < TOL$ . We do  $a = 3$  uniform refinements and  $b = 5$  additional local refinements. As convergence is guaranteed, if condition (13.16) holds, and since we know that in general  $\|\nabla u\|_{L^2} < Ch\|u\|_{L^2}$  for  $u \in \mathcal{U}$ , the step sizes are automatically set to  $\tau = \sigma = L * 2^{-(a+b)}$  with a constant  $L$ . For the  $\mathcal{L}^0/\mathcal{L}^0$  case, numerics indicate, that it is possible to even set  $\tau = \sigma = L * 2^{-(a+b)/2}$ . Note that the choice of  $L$  does not only depend on the operator norm of the gradient on the domain, but also on the number of elements in the initial grid. Here we choose  $L = 0.19$ . The overrelaxation parameter  $\theta$  is chosen to be 1.

By imposing all equations weakly the primal-dual algorithm de facto uses the  $L^2$ -projection  $\mathbb{P}g \in \mathcal{U}$  instead of the datum  $g \in \mathcal{L}^0(\mathcal{T}_h)$ . Figure 13.1 shows that in the case of  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$  by projecting a discontinuous function onto a continuous space we introduce an additional error in contrast to the case  $\mathcal{U} = \mathcal{L}^0(\mathcal{T}_h)$ . The over- and undershoots of the continuous projected data do not pose a problem, because they are regularized over the course of the algorithm, as we can see in Figure 13.2(b). Comparing Figure 13.2(a) with 13.2(b) and Figure 13.2(c) with 13.2(d) we see that choosing  $\mathcal{P} = \mathcal{S}^1(\mathcal{T}_h)^d$  smears out jumps, i.e., the discontinuities are less accurately approximated. This is due to the fact, that  $\mathcal{P}$  is continuous and its basis functions have a larger support than if it were discontinuous. Additionally  $\mathcal{P} = \mathcal{S}^1(\mathcal{T}_h)^d$  is not

### III. Finite Element Discretization of the Total Variation



**Figure 13.1:** The Projection of the piecewise constant datum  $g$  onto the space  $\mathcal{U}$ .

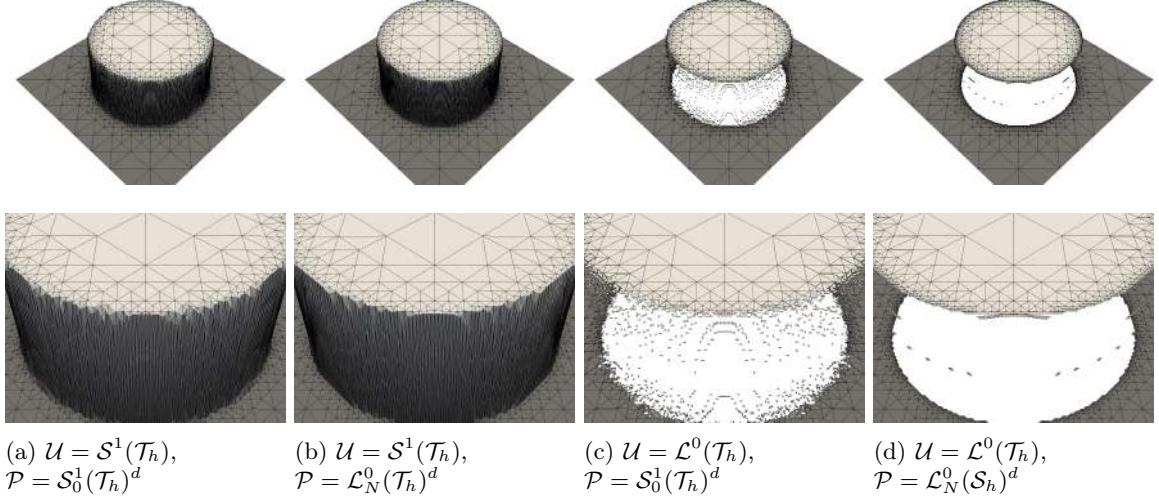
the space for which the primal and the mixed formulation are equivalent leading to a worse approximation of  $|Dv|(\Omega)$ . This explains why the value of the functional in Figures 13.3 and 13.4 for these variants is higher than choosing  $\mathcal{P} = \mathcal{L}_N^0(\dots)^d$ . In particular this worsens the quality of the solutions obtained with  $\mathcal{U} = \mathcal{L}^0(\mathcal{T}_h)$ ,  $\mathcal{P} = \mathcal{S}_0^1(\mathcal{T}_h)^d$ , as now multiple elements at the jump are hanging mid-air (Figure 13.2(c)), which drastically increases the total variation.

Figure 13.3 depicts the evolution of the energy  $J_{\alpha_1, \alpha_2}$  during the iterations for the considered space pairings. We observe that  $J_{\alpha_1, \alpha_2}$  is not monotonically decreasing but stagnates at a minimal value after a certain number of iterations (note that the scale of the number of iterations in Figure 13.3 is logarithmic). This demonstrates, that in all these settings the algorithm converges to a stationary point of the respective discrete problem. Due to the different combinations of spaces it is clear that the stationary points in general do not coincide and hence the minimal energy is different. The combination  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$ ,  $\mathcal{P} = \mathcal{L}_N^0(\mathcal{T}_h)^d$  yields the best result, as its final energy is the smallest among the considered cases.

We observe that choosing  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$  results in a decreasing energy for  $h \rightarrow 0$ , while for  $\mathcal{U} = \mathcal{L}^0(\mathcal{T}_h)$  this is not the case (cf. Figure 13.4). More precisely for the case  $\mathcal{L}^0(\mathcal{T}_h), \mathcal{S}_0^1(\mathcal{T}_h)^d$  the energy seems to oscillate in a certain sense and for  $\mathcal{L}^0(\mathcal{T}_h), \mathcal{L}_N^0(\mathcal{T}_h)^d$  the energy stays almost constant. This is due to the fact that, while the approximation of the circle gets better, the total variation does not diminish for  $\mathcal{L}^0$ . In general approximating functions of bounded variation is not possible with piecewise constant functions on a triangulation, since the length of discontinuities may not diminish over refinement [3]. This is different for the case  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$ , where discontinuities of functions, that are not representable on the triangulation, can be better approximated.

#### 13.4.2. Image Denoising

Here we demonstrate the denoising capability of the algorithm. Motivated by the above experiments we choose the spaces to be  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$ ,  $\mathcal{P} = \mathcal{L}_N^0(\mathcal{T}_h)^d$ . The considered image with values in  $[0, 1]$  is first corrupted by Gaussian white noise with variance 0.1 and then salt-and-pepper noise with  $s = 0.1$  is added. The obtained image is shown in Figure 13.5(a). In order to reconstruct the image we choose  $\alpha_1 = 250$ ,  $\alpha_2 = 150$  in (13.1), and perform our



**Figure 13.2:** The discrete minima  $u^*$  of the functional  $J_{10,20}$

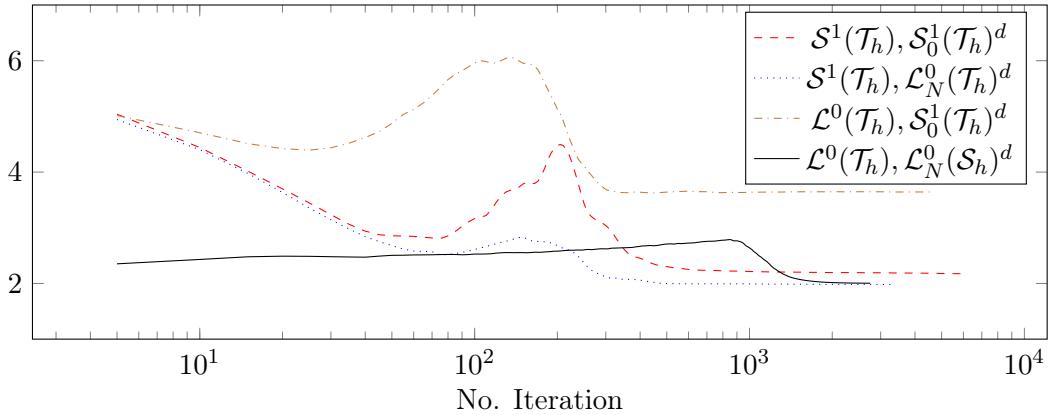
primal-dual algorithm with the tolerance set to  $10^{-2}$ ,  $\theta = 1$  and  $\sigma = \tau = 2^{-8}$ . To resolve the  $256 \times 256$  pixel-sized picture we apply 8 uniform refinements and no additional local refinement. In Figure 13.5(b) we depict the output of our algorithm. The result is reasonably smoothed due to the total variation regularization, while discontinuities are still preserved.

As a second example we run the algorithm on a much finer image, that has a resolution of  $2576 \times 1920$  pixels (Figure 13.6(a)). The spaces are chosen as above, the cost parameters are  $\alpha_1 = \alpha_2 = 500$ , we run the image on an initial grid of  $161 \times 120$  squares, each subdivided in two triangles and initiate 2 initial global refinements and 2 local Refinements. So locations with full refinement a triangle is half the size of a pixel. A part of the mesh is depicted in Figure 13.6(b), where one may observe that the discontinuities are captured by the refinement, as intended. The resulting image (Figure 13.6(c)) behaves as expected, the noisy structure of the stone is reduced to a minimum, but we also lose some details inside the stars, that we may have wanted to keep. This may be attributed to the choice of the parameters  $\alpha_1$  and  $\alpha_2$ . We note that in this example as well as in the previous one, these parameters are chosen at will, but not optimal. For an optimal choice of parameters, we refer the reader to [19]. Moreover it may be of interest to choose local cost parameters, as in [9, 12, 20].

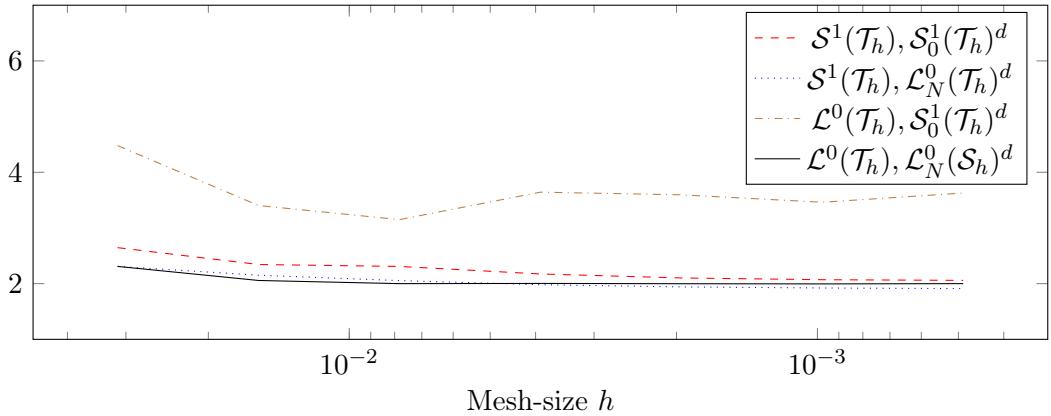
### 13.4.3. Strong Scaling

We do a strong scaling experiment on the same datum as in Section 13.4.1 using the spaces  $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$  and  $\mathcal{P} = \mathcal{L}^0(\mathcal{T}_h)^d$ . The computation is executed on a 32 core shared memory system. The grid manager DUNE-ALUGRID balances the computational load on the initial grid by partitioning it onto the different processors. So we cannot expect the algorithm to scale if the initial mesh is to coarse. E.g. if there are only two initial elements, only two processors can get partitions that are nonempty. Consequently we discretize the unit square by  $8 \times 8$  elements, see `data/finecube_2d.dgf`. As the grid is already fine, we do not need as many uniform refinements to reach a good resolution, so we do  $a = 3$  uniform and another  $b = 3$  local refinements. This results in a different operator norm of the gradient on the initial

### III. Finite Element Discretization of the Total Variation



**Figure 13.3:** The value of the functional  $J_{\alpha_1, \alpha_2}$  over the course of the algorithm.



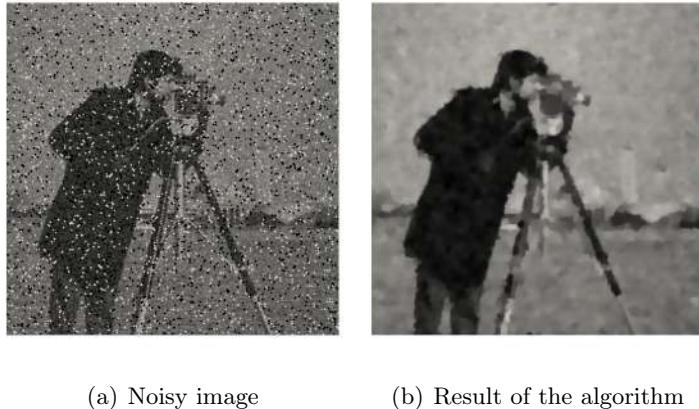
**Figure 13.4:** Final energy for  $h = 2^{-5}, \dots, 2^{-11}$

grid, so we have to set the corresponding parameter `nsm.constantL` to 0.02 for the stepsizes  $\tau = \sigma = L * 2^{-(a+b)}$  to be computed correctly. The tolerance is set to  $10^{-2}$  and  $\alpha_1 = \alpha_2 = 150$  to reach a short runtime.

Figure 13.7 indicates that the strong scaling at least up to 8 cores is quite good and afterwards it stagnates, which is probably due to the small grid and not enough workload. We would have to increase the number of initial cells to improve the scaling further. The important part about this result is, that the parallelism is done inside DUNE-ACFEM and we spent almost no effort to parallelize the code. The only line of code needed initializes MPI.

## 13.5. Summary and Outlook

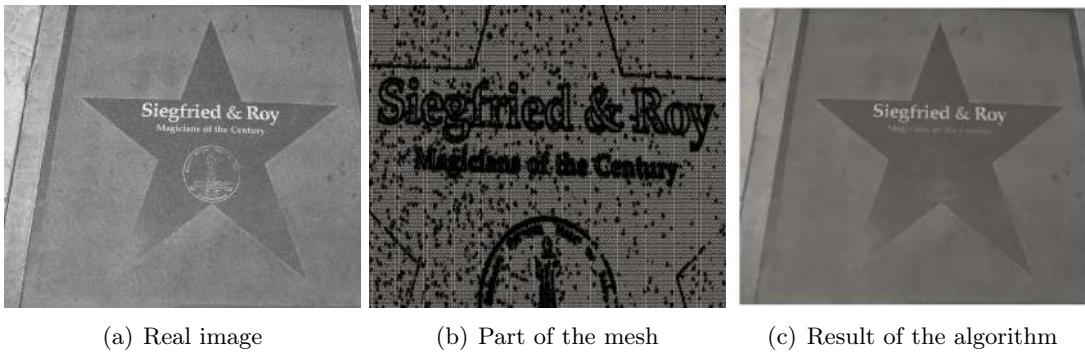
We have shown for a certain discretization that the associated minimizer converges to the continuous one. The primal-dual algorithm used here is implemented very conveniently within DUNE-ACFEM. The flexibility of DUNE-FEM allows us to easily exchange discrete spaces and easily set parameters. Also the algorithm is now intrinsically parallel by domain decomposition



(a) Noisy image

(b) Result of the algorithm

*Figure 13.5:* A corrupted image before and after applying the algorithm



(a) Real image

(b) Part of the mesh

(c) Result of the algorithm

*Figure 13.6:* A real image before and after applying the algorithm

with a decent strong scaling. Future research includes implementing a semismooth Newton method in DUNE-ACFEM to increase convergence speed even for highly adaptive grids.

## 13.6. Appendix A - Installation Instructions

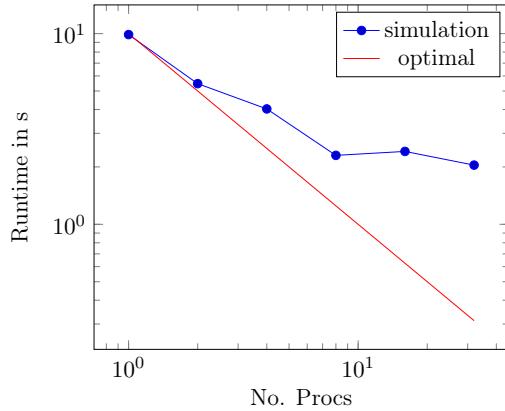
The program is designed to work in a Linux-environment. It may as well work on a unix machine. To install the program, first download the DUNE modules, DUNE-COMMON, DUNE-GEOMETRY, DUNE-GRID, DUNE-ISTL, DUNE-LOCALFUNCTIONS, DUNE-FEM, DUNE-ALUGRID and DUNE-ACFEM from [gitlab.dune-project.org](https://gitlab.dune-project.org) and `non-smooth-minimization` from <http://www.ians.uni-stuttgart.de/nmh/downloads>. For the DUNE- modules checkout the branch `releases/2.4`. Put all the projects in a directory as direct subfolders. Use your config file `config.opts` to run the command

```
./dune=common/bin/dunecontrol --opts=config.opts --module=non-smooth-minimization all
```

An example config file `example.opts` is provided in the main directory of the DUNE project `non-smooth-minimization`.

Now DUNE is installed and the executable `nsm` from the subdirectory `src` should have been

### III. Finite Element Discretization of the Total Variation



**Figure 13.7:** Strong scaling of the algorithm.

built within the `cmake` build directory. The scripts `example1.sh` to `example5.sh` produce the results presented in this paper. They require a `python` interpreter and `example3.sh` and `example5.sh` require `MPI`. The `cmake` build directory is assumed to be named `build-cmake`. This can be adjusted by modifying the `BUILDDIR` variable inside the scripts. The scripts are simple, so changes should be easy. They basically consist of changing directories, compiling the executable `nsm`, executing it with the right set of parameters and parsing the output into suitable directories. There are two types of output. Console output from `std::out` is parsed into a file `output.graph` in a subdirectory of `src`, which describes the value of the functional and the value of all its parts with respect to both  $g$  and  $\mathbb{P}g$  and a file `output.parameters`, where the applied parameters and other output can be looked up. Console output from `std::err` is displayed on the console. The other type of output is in the `cmake` build directory located within the directory `output` in suitable subdirectories. These contain a set of `.vtu/.vtk` files (readable with `Paraview`), that contain the datum  $g$  (called "image"), the projection  $\mathbb{P}g$  (called "projection of  $g$ "), the solution (called " $u0$ ") and the adjoint variable (componentwise " $p0$ ", " $p1$ "). It may take some time to run the scripts. To get faster results, simply edit the `LOCREF` or `INITREF` variable of the bash scripts or lower the tolerance parameter `nsm.tolerance`.

## 13.7. Appendix B - Parameters

There are two types of parameters: run time parameters and compile time parameters.

- Run time parameters can be overloaded on the command line by  
`./nsm nsm.parameter:value ...`.  
The default values are set inside the parameter file located at `data/parameter`. This file is copied into the `cmake` build directory at compile time. So parameters changed in the build directory will be overwritten when recompiling. Parameters with prefix `fem`, or `istl` belong to DUNE-FEM and DUNE-ISTL respectively, and are explained in their documentation. The specific parameters of this algorithm are prefixed `nsm`. The extensive list of run time parameters reads:

### 13. Using DUNE-ACFem for bounded variation functions

nsm.theta	Overrelaxation parameter $\theta$
nsm.maxIt	Maximum number of iterations of the algorithm
nsm.tolerance	The algorithm breaks if the tolerance is reached
nsm.outputStep	Output every nth step
nsm.constantL	If ! NSM_SET_STEPSIZE, $\tau = \sigma = L * 2^{-(a+b)}$
nsm.tau	Stepsize $\tau$
nsm.sigma	Stepsize $\sigma$
nsm.lambda_1	$L^1$ -data term cost coefficient $\alpha_1$
nsm.lambda_2	$L^2$ -data term cost coefficient $\alpha_2$
nsm.image	Filename of the image to be read in
nsm.initialRefinements	Number of initial uniform refinements a
nsm.localRefine	Number of initial adaptive refinements b
nsm.adaptTolerance	Adaptive tolerance
nsm.noiseType1	The CIMG type of noise to be applied first
nsm.noiseLevel1	The CIMG noise level of the first noise
nsm.noiseType2	The CIMG type of noise to be applied second
nsm.noiseLevel2	The CIMG noise level of the second noise

- Compile time parameters have to be declared when compiling. The location is in the file `CMakeLists.txt`. As they are `cmake`-cache variables, they can be redefined in the usual way (see e.g. `example1.sh`). They determine what kind of problem to treat and which discrete spaces are to be used.

NSM_SET_STEPSIZE	If true, $\tau$ and $\sigma$ are set manually
NSM_USE_IMAGE	If true, image is used instead of geometric expression
NSM_USE_NOISE	If true, noised imaged is used, requires NSM_USE_IMAGE
NSM_U_DISCONT	If true, $\mathcal{U} = \mathcal{L}^0(\mathcal{T}_h)$ , else $\mathcal{U} = \mathcal{S}^1(\mathcal{T}_h)$ .
NSM_P_DISCONT	If false, $\mathcal{P} = \mathcal{S}_0^1(\mathcal{T}_h)^d$ , else $\mathcal{P} = \mathcal{L}_N^0(\mathcal{T}_h)^d$ or $\mathcal{L}_N^0(\mathcal{S}_h)^d$ (depending on $\mathcal{U}$ ) .

## 13.8. References

- [1] M. Alkämper, A. Dedner, R. Klöfkorn, and M. Nolte. The dune-alugrid module. *Archive of Numerical Software*, 4(1):1–28, 2016.
- [2] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, 2000.
- [3] S. Bartels. Total variation minimization with finite elements: convergence and iterative solution. *SIAM Journal on Numerical Analysis*, 50(3):1162–1180, 2012.
- [4] S. Bartels. Broken sobolev space iteration for total variation regularized minimization problems. *IMA Journal of Numerical Analysis*, page drv023, 2015.
- [5] S. Bartels. Error control and adaptivity for a variational model problem defined on functions of bounded variation. *Mathematics of Computation*, 84(293):1217–1240, 2015.
- [6] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, R. Kornhuber, M. Ohlberger, and O. Sander. A generic grid interface for parallel and adaptive scientific computing. II. Implementation and tests in DUNE. *Computing*, 82(2-3):121–138, 2008.
- [7] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, S. Kuttanikkad, M. Ohlberger, and O. Sander. The Distributed and Unified Numerics Environment (DUNE). In *Proc. of the 19th Symposium on Simulation Technique in Hannover, Sep. 12 - 14*, 2006.
- [8] S. C. Brenner and R. Scott. *The Mathematical Theory of Finite Element Methods*, volume 15. Springer Science & Business Media, 2008.
- [9] C. V. Cao, J. C. De los Reyes, and C.-B. Schönlieb. Learning optimal spatially-dependent regularization parameters in total variation image restoration. *arXiv preprint arXiv:1603.09155*, 2016.

### III. Finite Element Discretization of the Total Variation

- [10] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [11] A. Dedner, R. Klöfkorn, M. Nolte, and M. Ohlberger. A Generic Interface for Parallel and Adaptive Scientific Computing: Abstraction Principles and the DUNE-FEM Module. *Computing*, 90(3–4):165–196, 2010.
- [12] Y. Dong, M. Hintermüller, and M. M. Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *Journal of Mathematical Imaging and Vision*, 40(1):82–104, 2011.
- [13] E. Giusti. *Minimal Surfaces and Functions of Bounded Variation*, volume 80 of *Monographs in Mathematics*. Birkhäuser Verlag, Basel, 1984.
- [14] C.-J. Heine. Dune-acfem documentation. <http://www.ians.uni-stuttgart.de/nmh/stage/documentation/software/dune-acfem/doxygen/>, 2014.
- [15] M. Hintermüller and K. Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM Journal on Applied Mathematics*, 64(4):1311–1333, 2004.
- [16] M. Hintermüller and A. Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed  $L^1/L^2$  data-fidelity in image processing. *SIAM Journal on Imaging Sciences*, 6(4):2134–2173, 2013.
- [17] M. Hintermüller and M. Rincon-Camacho. An adaptive finite element method in  $L^2$ -TV-based image denoising. *Inverse Problems and Imaging*, 8(3):685–711, 2014.
- [18] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.
- [19] A. Langer. Automated parameter selection in the  $L^1$ - $L^2$ -TV model for removing Gaussian plus impulse noise. *accepted by Inverse Problems*, 2016.
- [20] A. Langer. Automated parameter selection for total variation minimization in image restoration. *Journal of Mathematical Imaging and Vision*, 57(2):239–268, 2017.
- [21] D. Tschumperlé. The cimg library - c++ template image processing toolkit. <http://www.cimg.eu/reference/>.