On Variational Methods for Motion Compensated Inpainting Technical Report

François Lauze, Mads Nielsen
DIKU, Institute of Computer Science, University of Copenhagen
Universitetsparken 1, 2100 Kbh Ø
Denmark
{francois,madsn}@diku.dk

Abstract

We develop in this paper a generic Bayesian framework for the joint estimation of motion and recovery of missing data in a damaged video sequence. Using standard maximum a posteriori to variational formulation rationale, we derive generic minimum energy formulations for the estimation of a reconstructed sequence as well as motion recovery. We instantiate these energy formulations and from their Euler-Lagrange Equations, we propose a full multiresolution algorithms in order to compute good local minimizers for our energies and discuss their numerical implementations, focusing on the missing data recovery part, i.e. inpainting. Experimental results for synthetic as well as real sequences are presented.

Keywords. Image Sequences Restoration, Bayesian Inference, Optical Flow, Image Inpainting, Variational Methods.

1 Introduction

Since the birth of the cinema, more than a century ago, and the apparition of video in the 50's, a huge amount of Motion Picture material has been recorded on different types of media, including celluloid reels, magnetic tapes, hard disks, DVDs, flash disks...

Unfortunately, this type of material is subject to artifacts, already at image acquisition, or, for instance, degradations of the storage medium, especially for reels and tapes, caused by bad manipulation, incorrect storage conditions, or simply the "patina of time". These degradations can be of extremely many different types. For films, some of the problems encountered include, but are far from limited to, image flicker, noise, dust, real dust or a deterioration caused by the use of incorrect chemicals when developing a negative, for instance, missing regions and vertical line scratches, usually due to mechanical failures in the cameras or projectors, degradation of the chemicals, color fading, and even the degradation/destruction of the support, as it is the case for triacetate films, where excessive storage temperature and humidity causes the formation of acetic acid (vinegar), and this process is auto-catalytic: once started, it cannot be stopped. Nitrate cellulose based films are highly flammable. Many movie theaters were burnt down because of that, and their storage is critical because of explosion risks. Video also is not exempt of problems. The Telecine, a device used to convert a film to video, can introduce artifacts such as distortions or Moiré patterns. Repeated playbacks may damage the tape, introducing for instance noise, the lost of synchronization information produces a disturbing jittering effect. The digital medium has also its own problems, such as the block artifact due to lossy compression methods, and during network transmission, some blocks can be lost.

Among these degradations, *blotches* – i.e, following Kokaram ([33] "regions of high contrast that appear at random position in the frame (...)". In these regions, the original data is usually entirely lost and this paper focuses on recovering the missing data. Digital Inpainting is the name that is commonly given to the process of filling-in regions of missing data in an image, and although the term was first coined for spatial images in [9], Image Sequence or Video Inpainting is now widely used. A main difference resides

in the fact that information missing in one frame is supposed to exist in adjacent frames, this is used to restrict the search space, via interframe motion estimation.

With the increasing computational power of modern computers, more and more sophisticated algorithms are developed for inpainting of still frames as well as video sequences. They are often divided into stochastic and deterministic methods, although the two approaches often share many modeling aspects.

Spatial Image Inpainting techniques were pioneered by Masnou and Morel [38], location where data is missing is treated as an object occluding the background and a Elastica based variational model for the completion of broken level lines is proposed. Bertalmio *et al.* [9] proposed a Partial Differential Equations (PDE) based method for filling-in regions by a sort of smoothness transport. They proposed then a variational technique for recovery of level lines and intensity in [7]. Chan and Shen proposed in [17] approaches based on Total Variation as well as Elastica minimization. Using a different, discrete approach, inspired by work on texture generation [22, 21], Criminisi *et al.* proposed a simple and elegant patch based method in [19].

Apart from the deterministic/non deterministic division, there exists another level of modeling, One might speak of a low level analysis approach for a direct study of the image geometry in the spatial case, or a "simple" 2D+time model, while an *high level analysis* approach will attempt to model/infer key descriptions of the underlying 3D scene and/or its dynamics using for instance tracking, hypotheses of (quasi)-repetitive motion , and use them for sequence reconstruction.

In the latter category, Irani and Peleg in [29] proposed a method that can among others remove foreground objects from a scene. From a multilayer parametric motion representation, foreground occluded objects can be detected and removed thanks to the motion and intensity information available for the other layers.

Extending the spatial exemplar based inpainting work of Criminisi *et al.* in, Patwardhan *et al.* have proposed techniques for video inpainting, in the case of static background [44] and extended to relatively simple camera motion [45].

In the other hand, a series of more "low level" methods have been able to produced high quality results. Kokaram, in a series of works [35, 33, 34] explored probabilistic "2D+1" based approaches for joint detection and reconstruction of missing data. Grossauer in [28] used optical flow in order to detect serious violations which might then be interpreted as blotches, and then performs a simple transport from neighbor frames. In a similar way, D'Amore *et al.* [20]) also developed methods for detection and removal of blotches based on variational motion estimation. In a different way, Wexler *et al.* [51] proposed a patch based method for filling in holes, with constrains enforcing spatial en temporal coherence. In a recent work, Wang and Mirmehdi [49] used spatiotemporal random walks in order to determine best replacement pixels, taking into account surrounding values.

The work we present here can be seen as a deterministic "lower level" approach to the blotch restoration problem, where we are given a damaged image sequence, as well as the location of the damages, assumed to have been obtained by specific means. Some recent automatic approaches provide good results, see for instance Buades *et al.* [15] or the one of Wang and Mirmehdi [48]. In the other hand, precise detection of large blotches can be difficult, and semi-automatic procedures are very often applied.

Starting from a general Bayesian Inference approach, and using standard approaches in motion recovery, we derive a generic variational formulation for joint estimation of motion and recovery of missing data, we then instantiate this generic class into several energy functionals, we propose to solve them via their Euler-Lagrange Equation in a fully multiscale framework.

Some of these ideas appeared also in the monograph of Aubert and Kornprobst [6], as a contribution from the first author of this paper. We have also used variations over this framework for Video Deinterlacing in [31], temporal super-resolution [32] and spatial super-resolution [30]. Ideas presented here where exposed by the authors in a conference paper [36], and turned to be similar to the ones of Cocquerez *et al.* [18], but were developed independently. The connection between these two works was pointed out to the authors by R. Deriche during a stay of the first author at INRIA Sophia-Antipolis in 2003.

This paper is organized as follows. After having introduced some notations in the first section, a probabilistic model is discussed in section 2, modeling both the purely spatial behavior for a still frame extracted from a video sequence, as well the temporal correlation which is expected between the frames, correlation coming from the *apparent motion*. More precisely, what the model describes is the joint probability of an image sequence and a vector field describing apparent motion, knowing that they

should agree with a given degraded image sequence. Using Bayesian inference, an expression for a posterior probability distribution is worked out. Then following Mumford in [41], a generic family of variational formulations is deduced for the problems of simultaneous motion recovery and inpainting, as well as simultaneous motion recovery, inpainting and denoising. Minimizations for these variational formulations provide our *variational motion compensated inpainting* algorithms.

2 Bayesian Framework for Inpainting and Motion Recovery

In this section we introduce a generic probabilistic framework for the missing data problem, via Bayesian inference. We first provide a general rationale for writting a posterior probability and via maximum a posteriori estimation, we derive a generic energy minimization for the inpainting problem.

2.1 A Posterior for missing data

We use discrete settings in order to easily give a sense to the different probability computations used. Given the degraded sequence u_0, we assume the existence of a degradation process $\mathcal{P}$ which results in u_0, this can be the loss of every odd / even lines in an alternating way, as it is the case in video interlacing, a spatial or spatiotemporal downsampling or as a less structured collection of missing blocks of pixels, described by their location Ω with the sequence's spatial domain D. Our goal is to reconstruct a sequence u on D and to compute a motion field $\vec{v}$ for that sequence. We introduce therefore the conditional probability $p(u, \vec{v}|u_0, \mathcal{P})$. Using The Bayes rule of retrodiction, we can write this conditional probability as a product of a *likelihood* and a *prior* divided by an *evidence*

$$p(u, \vec{v}|u_0, \mathcal{P}) = \frac{p(u_0|u, \vec{v}, \mathcal{P})\, p(u, \vec{v}|\mathcal{P})}{p(u_0|\mathcal{P})}. \tag{1}$$

The evidence corresponding to the known data plays the role of a normalizing constant and will be disregarded in the sequel. We assume that u_0 is a degraded version of u, with for instance noise added in the acquisition process or because of aging, and missing data coming from film abrasion due to a mechanical failure or a bad manipulation. Therefore the observed degraded image sequence u_0 is assumed to depend only on u and not on the apparent motion $\vec{v}$ – i.e., occlusion and desocclusion are not considered as degradations, they "belong" to the original sequence. The likelihood term $p(u_0|u, \vec{v}, \mathcal{P})$ is therefore just $p(u_0|u, \mathcal{P})$. Because of the nature of the causes for missing data, we can assume the independence of $(u, \vec{v})$ and $\mathcal{P}$ and the prior is simply

$$p(u, \vec{v}|\mathcal{P}) = p(u, \vec{v}) = p(u|\vec{v})\, p(\vec{v}).$$

We will now decompose this prior. For that, let's imagine the following common situation. Using a DVD player, a person is watching a video on a TV set. He/she pushes the "pause" button of the DVD player's remote control. There could be motion blur due to large motion and finite aperture times at image acquisition, but our spectator expects to see a meaningful still image displayed on the TV screen. When the "play" button is pushed again, the animation resumes, and this person expects to see an animation which is coherent with the content of the still, at least for a sufficiently small amount of time, this coherence corresponding of course to the *apparent motion*. This leads us to assume that $p(u|\vec{v})$ has the form $p(u_s, u_t|\vec{v})$ where u_s and u_t denote local spatial (still frame) and temporal (animation) distributions for u, and we factor it as

$$p(u_s, u_t|\vec{v}) = p(u_t|u_s, \vec{v})\, p(u_s|\vec{v}). \tag{2}$$

As we see, the first term of this factorization is a direct translation of this expected temporal coherence. For sake of simplicity, we assume the independence of u_s and $\vec{v}$. This is not necessarily true, as motion edges and image edges have a tendency to be correlated, a fact exploited by several motion recovery algorithms, starting with the work of Nagel and Enkelmann (see for instance [42, 3, 2]). On the other hand, many optical flow algorithms do not use this potential dependency and provide very accurate motion estimations, in fact often better as approaches that take explicitly into account spatial edges have

a tendency to produce over-segmented flows (see [39] for instance). Putting all these elements together, we finally obtain the following posterior distribution

$$p(u,\vec{v}|u_0,\mathcal{P}) \propto \underbrace{p(u_0|u,\mathcal{P})}_{P_1}\ \underbrace{p(u_s)}_{P_2}\ \underbrace{p(u_t|u_s,\vec{v})}_{P_3}\ \underbrace{p(\vec{v})}_{P_4} \tag{3}$$

where P_1 is the likelihood of u, P_2 is the spatial prior for the sequence, P_4 is the motion prior, and P_3 is a coupling term that acts both as a temporal prior for the sequence and a likelihood for the motion – as is the gray level constancy assumption along apparent motion trajectories seen from either image or motion point of view.

3 Variational Formulation

Among others, a standard way to compute a pair $(\bar{u},\bar{\vec{v}})$ from (3) is to seek for the *maximum a posteriori* (MAP) of this expression. The kind of probability functions which are normally used are *Gibbsian*, i.e. of the form $(1/Z)e^{-E}$ where E is expressed as a sum over cliques, ([26]) and stochastic algorithms can be used, as for instance Markov Random Field techniques.

From this point we will assume that the degradation process is given as a known missing data locus $\Omega \subset D$, the spatiotemporal domain of the sequence, and the reconstruction will be though as blotch removal, although much of what is presented below will still formally be valid for deinterlacing or super-resolution. Before discussing this, we introduce some notations and concepts that will be used in the sequel.

3.1 Notations

The approach for sequence modeling taken in this work follows ideas of ordinary differential equations and dynamical systems. The spatial domain of a sequence will will be taken to be the unit square $D_s := (0,1)^2 \in \mathbb{R}^2$. The spatial coordinates will be denoted $\mathbf{x} = (x,y)$, the temporal coordinate by t and spatiotemporal coordinate $r = (\mathbf{x},t)$. Given $A \subset B$ the complement of A in B will be denoted by $B\backslash A$ or just A^c (when B is clear from the context). $D_t = [0,T]$, $D_t^- = [0,T)$, $D_t^+ = (0,1]$. $D = D_s \times D_t$, $D^- := D_s \times D_t^-$, $D^+ := D_s \times D_t^+$. A sequence will be a map $u : D \to \mathbb{R}^n$, with typically $n = 1$ for gray value sequences and $n = 3$ for color ones. As used above, Ω will denote the subset of D of missing data of a sequence. We assume that D is made of "planar particles" $\mathbf{x}(t)$ that move smoothly in D, at least locally, out of some codimension 2 regions, with velocities $d\mathbf{x}/dt = \vec{v}(\mathbf{x}(t)) = \vec{v}(\mathbf{x},t)$ the instantaneous spatial velocity vector field. To each spatial velocity field $\vec{v} = (v_1,v_2)^t$, we associate a spatiotemporal velocity field $\vec{V} = (v^t,1)^t$.

In this setting we consider instantaneous vector fields $\vec{v} : D \to \mathbb{R}^2$. They will not a priori be a displacement, but a velocity, i.e. the spatial part of the time-derivative of a particle trajectory.

Given a smooth vector field $X : D \to \mathbb{R}^3$ smooth enough, The directional or Lie derivative of a function f in the direction X at r is simply

$$\mathcal{L}_X f(r) = X(r) \cdot \nabla_3 f_{(r)}$$

(∇_3 denotes the *spatio-temporal gradient*)[1]. Standard results on ordinary differential equations show that X has a *local flow* c^X, i.e. a local solution of the differential equation $\dot{c}^X = X \circ c^X$ and that for h small enough the map θ_r^X, $h \mapsto c^X(r,h)$ is a local diffeomorphism from an open subset of D into D (see [25] for a detailed exposure). Then on has

$$\mathcal{L}_X f(r) = \lim_{h\to 0} \frac{f(\theta_r^X(h)) - f(r)}{h}. \tag{4}$$

It makes clear that if $h > 0$,

$$\mathcal{L}_X^{h+} f(r) := \frac{f(\theta_r^X(h)) - f(r)}{h} \tag{5}$$

[1] when f is vector-valued, $\nabla_3 f$ can be taken to be $(Jf)^t$, the transpose of the Jacobian matrix of f and $\nabla f \cdot X := Jf\,X$

is a forward differences approximation of $\mathcal{L}_X$, and it we will use when developing discretizations for numerical solvers, as well as we will use its *backward difference* approximation

$$\mathcal{L}_X^{h-} f(r) := \frac{f(r) - f(\theta_t^X(-h))}{h}. \tag{6}$$

We will often forget the h superscript and only write $\mathcal{L}_X^-$ and $\mathcal{L}_X^+$. Given a "particle" $\mathbf{x}$ at time t, under the velocity field $\vec{v}$, its position at time $t+h$ is given by $\theta^{\vec{V}}_{(\mathbf{x},t)}(h)$, where $\vec{V}$ is as above the spatio-temporal extension of $\vec{v}$, and this position has the form $(\mathbf{y}, t+h)$ and thus induces a *spatial displacement field*

$$\mathbf{v}(\mathbf{x}, h, t) = \mathbf{y} - \mathbf{x}, \tag{7}$$

obtained by integrating the values of $\vec{v}$ in the interval $[t, t+h]$. We will denote its spatiotemporal extension by $\mathbf{V}$.

For a spatial velocity field $\vec{v}$, we will denote, abusively, by $\mathcal{L}_{\vec{v}}$ what should be $\mathcal{L}_{\vec{V}}$, and similarly for $\mathcal{L}_{\mathbf{v}}^{h+}$ and $\mathcal{L}_{\mathbf{v}}^{h-}$. Denoting by ∇ the *spatial* gradient operator, one has

$$\mathcal{L}_{\vec{v}} f = V \cdot \nabla_3 f = \vec{v} \cdot \nabla f + f_t.$$

We will use spatial and spatio-temporal divergence operators, respectively denoted $\nabla\cdot$ and $\nabla_3\cdot$, although, we will generally drop the subscript in the last notation, as the dimension of the vectorial argument of $\nabla\cdot$ should remove ambiguity. For a map $f : D \to \mathbb{R}^k$, $J(f)$ will denote the Jacobian of f with respect to its spatial variables and for a map $g : D \to \mathbb{R}$, $\mathcal{H}(g)$ will denote its Hessian with respect to spatial variables.

To end this section, we will denote by χ_A the characteristic function of a set A:

$$\chi_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}.$$

3.2 From MAP to Minimization

The MAP problem is first transformed into a discrete energy minimization problem

$$\begin{aligned} E(u, \vec{v}) &= -\log(p(u, \vec{v}|u_0, \Omega)) = \lambda_1 E_1(u) + \lambda_2 E_2(u_s) \\ &\quad + \lambda_3 E_3(u_s, u_t, \vec{v}) + \lambda_4 E_4(\vec{v}) \end{aligned} \tag{8}$$

with $\lambda_i E_i = -\log(P_i)$ $(\lambda_i > 0)$, and our goal is thus to find a minimizer $(\bar{u}, \bar{\vec{v}})$ of $E(u, \vec{v})$, E being a function of the image pixel locations and flow locations through u and $\vec{v}$. When image and motion field spaces are provided Euclidean structures, and E is reasonably smooth, a necessary (but not sufficient) condition for $(\bar{u}, \bar{\vec{v}})$ to be a minimizer is to be a zero of the energy gradients with respect to u and v, gradients that represent the corresponding differentials for the corresponding Euclidean structures, and denoted by $\nabla_u E$ and $\nabla_v E$ in the sequel:

$$\begin{cases} \nabla_u E(\bar{u}, \bar{\vec{v}}) = 0 \\ \nabla_{\vec{v}} E(\bar{u}, \bar{\vec{v}}) = 0. \end{cases} \tag{9}$$

Note however that expression (8) makes sense only when we can take the log of each of the P_i distributions. An important situation where this is clearly not the case when the likelihood term has the form of a Dirac distribution:

$$P(u_0|u) = \delta(Ru - u_0) = \begin{cases} 1 & \text{if } Ru = u_0 \text{ on } D\backslash\Omega \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where R is an operator such as a spatial or spatiotemporal blur, or simply the identity. In that situation, we simply seek for a $(\bar{u}, \bar{\vec{v}})$ satisfying

$$\begin{cases} (\bar{u}, \bar{\vec{v}}) & = \underset{(u,\vec{v})}{\text{Argmin}}\, \lambda_2 E_2(u_s) + \lambda_3 E_3(u_s, u_t, \vec{v}) + \lambda_4 E_4(\vec{v}) \\ Ru = u_0 & \text{on } D\backslash\Omega. \end{cases} \tag{11}$$

In that case, a minimizer $(\bar{u}, \vec{\bar{v}})$ should satisfy the following conditions

$$\begin{cases} \nabla_u E(\bar{u}, \vec{\bar{v}}) = 0 & \text{on } \Omega \\ R\bar{u} = u_0 & \text{on } D\backslash\Omega \\ \nabla_{\vec{v}} E(\bar{u}, \vec{\bar{v}}) = 0 \end{cases} \tag{12}$$

At this point, the formulation we have obtained is discrete in nature. Using the Bayesian to variational formulation proposed by Mumford in [41], limiting expressions for the probability distributions/energy involved give rise to a *continuous* energy formulation, treated at least at a formal level, and deterministic algorithms can be used to carry out the minimization problem. The continuous expressions will also be denoted E_i. The "game" is therefore to choose meaningful expressions for each of the E_i, which are also computationally *reasonable*. Variational 2-dimensional formulations of inpainting can provide the spatial term E_2 and variational optical flow algorithms can be "plugged-in" for terms E_3 and E_4, and really many of them have been proposed in the two last decades. We will extend upon these choices.

In the following paragraphs we argue on the instantiations of the different terms $E_1(u; u_0)$, $E_2(u)$, $E_3(u, v)$ and $E_4(v)$, they should present a good trade-off between accuracy of modeling and simplicity in order to result in computationally tractable algorithms.

3.3 Image data term

The data term provides a measure of deviation between an observed sequence and a candidate reconstruction. As mentioned above, this term is relevant in the inpainting-denoising case, and has generally the form

$$E_1(u; u_0) = \int_{\Omega^c} L(Ru, u_0)\, dx\, dt$$

For an additive noise, $L(x, y) = \phi(x - y)$, for a multiplicative one, $L(x, y) = \phi(y/x)$. Noise can be very complicated to apprehend, and generally a simple model is assumed, such as additive Gaussian white noise with a fixed variance or a more general Laplace noise, and $\phi(x - y) = |x - y|^p$ for a $p \geq 1$. In this work we have restricted ourselves to the Gaussian white noise case: $\phi(x - y) = (x - y)^2$ and assume that $R = \mathbb{I}$, the identity transform – we ignore blurring, or merely we assume blurring is part of the sequence to be restored.

3.4 Spatial regularization terms

Such a term would generally have the form

$$E_2(u) = \int_D L(\nabla u, \nabla^2 u, \dots)\, dx\, dt$$

where we recall that ∇ denotes the spatial gradient. Derivatives of order > 1 are rarely used in practice (although see [37]). Tikhonov regularization corresponds to the case $L(w) = \phi(|w|^2)$ with $\phi(x) = x$, but is to be avoided in general, as it blurs edges. One of the most popular choice is $\phi(x) = \sqrt{x}$ which corresponds to the Total Variation of Rudin, Oscher and Fatemi [46]. Because of its non-differentaibility at 0, is is often replaced by $\phi(x) = \sqrt{x + \epsilon^2}$ where $\epsilon > 0$ is small. Other ϕ, some non convex, have been reported to behave very well [6].

3.5 Motion likelihood – Temporal regularization

As our little story illustrated it, some form of coherence on the sequence content must be present. It is expressed as the requirement that some derived quantity from the sequence is conserved across time, quantities that depend on scene illumination, the image formation process, etc... In the temporal discrete setting, this conservation can be written, using forward differences along a motion field as

$$\mathcal{L}^+_{\mathbf{V}} F(u)(r) = 0, \quad \forall r \tag{13}$$

and in the continuous temporal setting, using the Lie-derivative

$$\mathcal{L}_{\vec{v}}(F(u)) = \vec{v} \cdot \nabla F(u) + \frac{\partial F(u)}{\partial t} = 0. \tag{14}$$

The continuous equation is generally derived as an approximation of the discrete one. In this work we used a view point a bit different: integrating along fixed time ranges the continuous equation will provide the discrete one.

The conservation equations (13)–(14) do not generally hold perfectly, due to a series of factors: among others, occlusions / disocclusion. Noise can also be a source of error, although presmoothing can be built into F. F can be vectorial in order to express that several elementary quantities are expected to be conserved, and will always be assumed to be a linear differential (or integro-differential) operator.

In order to cope with conservation equation failure, the conservation properties are enforced in a least square or generalized least square sense via terms

$$E_3(u, \vec{v}) = \int_D L\left(\mathcal{L}_{\vec{v}} F(u(r))\right) \, dr.$$

For an x seen as "a collection of features" $(x_1, \ldots, x_k)$ where the x_i's can themselves be vectors, we will always assume that $L(x) = \phi(\sum_i |x_i|^2)$ or $L(x) = \sum_i \phi_i(|x_i|^2)$ where $|x_i|$ is the standard Euclidean norm of x_i.

The most commonly used operator F is simply the identity: the intensities should match along the motion trajectories. Although clearly limited in validity [47], it works well in many cases. Higher order differential operators have been used, such as the spatial gradient, spatial Laplacian, spatial Hessian *etc.* These can be potentially attractive to enforce specific properties for motion recovery. The spatial gradient will prefer local translational motion. The Laplacian, being rotationally invariant would not penalze rotational motion... However one must keep in mind that a differential operator of degree n will give rise to terms of degree up to and including $2n$ in the Gâteaux derivative of E_3 wrt u. For that reason, we have limited our investigations to the identity and spatial gradients.

3.6 Motion regularization

A large amount of work has been devoted to regularizer terms for optical flow algorithms to accommodate the different situations encountered in practice, ranging from natural image sequences, medical image sequences, fluid dynamics data... Having Film/Video restoration as our goal, we focus here on natural image sequences. They are generally composed of moving rigid/semi-rigid objects projected on the camera plane, resulting mostly in locally translational motion.

Thus a regularizing term should enforce this behavior, in order to recover essentially smooth small varying regions with in general 1D discontinuities between these regions. We will limit ourselves to terms that depend on the spatial or spatiotemporal Jacobian of the motion field. More details will be provided in the next paragraph.

3.7 Proposed variational formulations

From the discussion above, we propose several actual variational formulations for the joint motion recovery / image sequence inpainting. For each formulation, we consider the inpainting/denoising problem and the pure inpainting formulation. For inpainting denoising we place ourselves in the additive Gaussian noise situation discussed in Subsection 3.3:

$$E_1(u; u_0) = \frac{1}{2} \int_{\Omega^c} (u - u_0)^2 \, dr. \tag{15}$$

The spatial regularization term used in this study is a regularization of the Total Variation term:

$$E_2(u) = \int_D \phi(|\nabla u|^2) \, dr \tag{16}$$

where $\phi(x^2) = \sqrt{x^2 + \epsilon^2}$ for a certain $\epsilon > 0$. From now, ϕ will always denotes this function. This type of regularization in the spatial case is known to generate starcaising effects artifacts [43]. Experiments in Section 6 on *Inpainting/Denoising* shows some tendency to it, when the weight of spatial regularization is relatively important. In pure inpainting, however, this term merely acts as brightness regularization and the above mentioned artifacts are much less stronger, on invisible.

We present candidates for remaining two terms in the following two tables. Table 1 deals with the motion likelihood/temporal regularization part while Table 2 deals with motion regularization terms. All these terms are "borrowed" from well-known variational optical flow formulations, λ_3 and λ_4 being some weights > 0. In the sequel, we will refer to one of these terms as E_i^k where E_i^k is the k-term (k-row) of the E_i array ($i = 3, 4$) presented below.

	$E_3(u, v)$
1	$\int_{D^-} \phi\left((\mathcal{L}_{\vec{v}} u)^2\right) dr$
2	$\int_{D^-} \phi\left((\mathcal{L}_{\vec{v}} u)^2 + \gamma \lvert\mathcal{L}_{\vec{v}} \nabla u\rvert^2\right) dr$
3	$\int_{D^-} \phi\left((\mathcal{L}_{\vec{v}} u)^2\right) + \gamma\phi\left(\lvert\mathcal{L}_{\vec{v}} \nabla u\rvert^2\right) dr$

Table 1: Terms coupling intensity and motion.

	$E_4(v)$
1	$\int_{D^-} \left[\phi\left(\lvert\nabla \vec{v}_1\rvert^2\right) + \phi\left(\lvert\nabla \vec{v}_2\rvert^2\right)\right] dr$
2	$\int_{D^-} \phi\left(\lvert\nabla \vec{v}_1\rvert^2 + \lvert\nabla \vec{v}_2\rvert^2\right) dr$
3	$\int_{D^-} \phi\left(\lvert\nabla_3 \vec{v}_1\rvert^2 + \lvert\nabla_3 \vec{v}_2\rvert^2\right) dr$

Table 2: Terms for motion field regularization

Terms E_3^1 and E_4^1 are regularizations of the 1-norm terms proposed by Aubert *et al.* ([4, 5]). Terms E_3^2 and E_3^3, in their time discrete version where $\mathcal{L}_V^+$ is used instead of $\mathcal{L}_v$, have been respectively proposed by Brox *et al.* [11] (that also proposed regularizer E_4^3) and Bruhn and Weickert [12] and proved to provide highly accurate flow. The time-continuous version E_3^2 was used in the authors' conference paper [36]. E_4^2 was proposed by Weickert and Schnörr in [50].

All the combinations were implemented, although we present only some of them in our experiments, for obvious space reasons.

We have not yet discussed the fact that image sequences can be vector-valued. For a sequence $u = (u_1, \ldots, u_n) : D \to \mathbb{R}^n$, $n > 1$, the same energy terms are used, where we then set

$$|\nabla u|^2 = \sum_{i=1}^{n} |\nabla u_i|^2, \quad (\mathcal{L}_{\vec{v}} u)^2 = \sum_{i=1}^{n} (\mathcal{L}_{\vec{v}} u_i)^2, \ldots \tag{17}$$

4 Resolution Methodologies

Minimizing the energies presented in the previous section is done via solving their corresponding flow and image Euler-Lagrange equations. Motion estimation requires normally a multiresolution / multiscale approach in order to be able to handle large displacements. We adopt a multiresolution approach here and we propose to interleave the images inpainting steps with the flow recovery ones at each resolution level. In this section we describe in more details this resolution approach and we introduce tools for computing the image Euler-Lagrange equations associated with our formulations. We then process to the computation of the different energy gradients, at least for the image related terms. Indeed, the steps for solving motion recovery are well studied and we follow the references we have mentioned in the previous section, we won't provide the details of their derivations.

4.1 Multiresolution framework

Motion computation depends critically on the range of the different discrete filters supports used for the estimation of the differential/finite differences quantities in the optical flow equation, and/or the smoothness of the input image data. For that reason multiscale or multiresolution techniques are needed in order to avoid meaningless local minima. On the other hand, a good approximation of the motion field at a given resolution should allow a precise recosntruction of image information at that resolution by motion compensated diffusion. Therefore we present here a methodology that aims at producing algorithms converging to a reasonable minimizer by solving iteratively for the image and the flow, in a complete *spatial* multiresolution setting. We implicitly build pyramids for the image sequence and the flow. In the sequel we assume that we have $N+1$ resolution levels, level N being the coarsest, level 0 being the finest, original one. We do not require a $1/2$ spatial scaling factor between consecutive levels.

We are seemingly in presence of an hen and egg problem, since in order to compute a flow with our methods we need image values, that are a priori unknown inside the degradation locus, and we need a flow in order to inpaint inside the degradation locus. This will be handled by using a degenerated form of the inpainting algorithm at coarsest resolution, that allows us to recreate a starting image sequence u^N. At coarsest resolution, the interframe motion is assumed to be very small, and we inpaint the sequence by computing a minimizer of the corresponding energy formulation where we have assumed $\vec{v} = 0$ – no motion is present. The non-linearity of the image equation provides a form for motion adaptivity.

Once we have a coarsest resolution inpainted sequence, we can start our multiresolution process. What we do is to modify the motion field recovery algorithm by running the inpainter after the interpolation of the lower resolution flow, and before updating the flow at that resolution level. Figure 1 illustrates a typical iteration. At a given pyramid level $k+1$, let us assume that we have an inpainted sequence u^{k+1}. It is then used, together with an interpolated version of $\vec{v}^k$, to compute a motion field at that resolution, to produce $\vec{v}^{k+1}$. Then at resolution level k, we first interpolate the flow, using for instance a bilinear or bicubic interpolation, to produce an *intermediary* flow field $\vec{v}_i^k$. This flow is used in order to reconstruct the sequence u^k at level k.

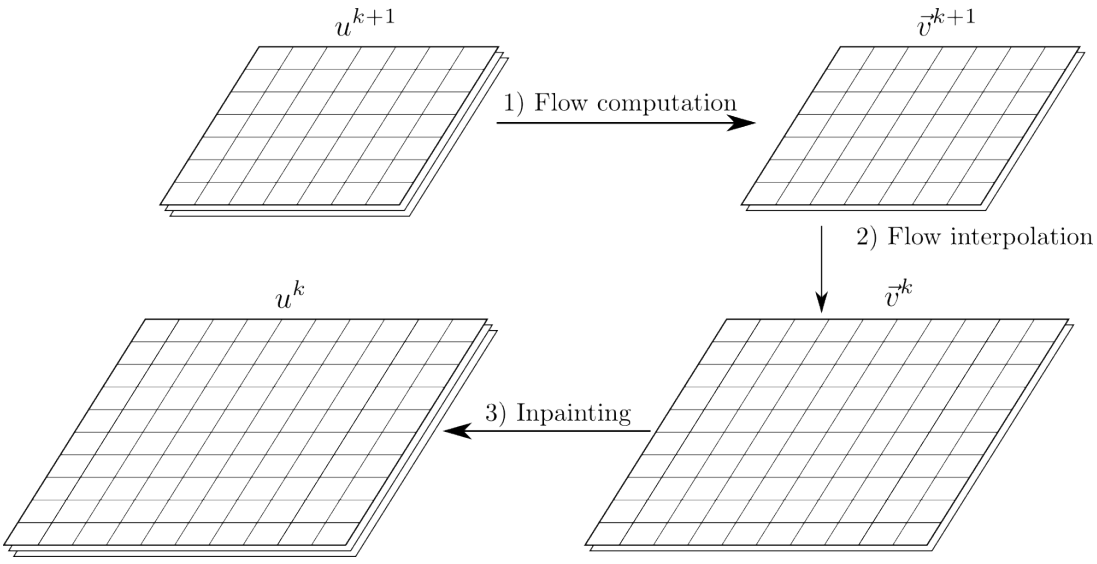


Figure 1: Multiresolution inpainting. From a sequence u^{k+1} at resolution level $k+1$, 1) the optical flow is computed, 2) it is interpolated at resolution level k and 3) used to inpaint the sequence at level k. This process is then iterated at lower pyramid levels.

A general algorithm (not tied up to our particular formulation) is given in table 3. As it can be seen from this sketchy algorithm, the difference between a generic multiresolution optical flow algorithm and our inpainter are simply the *interleaved* inpainting steps.

1. Compute u^N by spatial TV and motion adaptive diffusions.
2. For $k = N-1$ downto 1
   (a) Compute $\vec{v}^{k+1}$ from u^{k+1}
   (b) Compute intermediary flow $\vec{v}_i^k$ by interpolation of $\vec{v}^{k+1}$
   (c) Inpaint u^k using $\vec{v}_i^k$
3. Compute $\vec{v}^0$ from u^0
4. Output $(u^0, \vec{v}^0)$.

Table 3: Generic Motion Recovery and Inpainting Algorithm,

4.2 Derivation of Euler-Lagrange equations

In this subsection we compute the different energy derivatives with respect to image variables for the terms proposed in Subsection 3.7. The velocity related derivatives will be provided without calculation as they have been derived by authors mentioned in Subsection 3.7, at the exception of the new flow smoothing term. We first provide the general methodology to compute these derivatives and some elementary results used in the sequel. We restrict ourselve to scalar valued images and will indicate extension to the vector valued cases.

4.2.1 Computing functionals differentials

In order to obtain the Euler-Lagrange equations, we use the technique of directional derivatives for computing differentials dE_u of $u \mapsto E(u)$:

$$\xi \mapsto dE_u\xi = \ell'(0), \quad \ell(\tau) = E(u + \tau\xi) \tag{18}$$

and use of adjunction/ integration by part will allow to transform $dE_u\xi$, which generally appears as an inner product $\int_{\mathcal{D}} L_1(u)L_2(\xi)\,dx$ of a term L_1 involving derivatives of u and a term L_2 linear in ξ and some of its derivatives on the domain $\mathcal{D}$, into an equivalent expression

$$dE_u\xi = \int_{\mathcal{D}} F_u\xi\,dx + \int_{\partial\mathcal{D}} \xi G_u(\nu)\,ds$$

where $\partial\mathcal{D}$ is the boundary of $\mathcal{D}$, ν its exterior normal field, and under proper conditions on acceptable deformations ξ or on u on the boundary $\mathcal{D}$ making this boundary integral vanish, F_u is the *gradient* of E at u for this inner product. We will perform computations on each energy part and keep track of resulting boundary integrals.

We will present integration by part formulas for the spatial gradient operator ∇ as well as $\mathcal{L}_V$. They all result from the standard integration by part / Stokes formula [23].

Proposition 4.1 *Let $\mathcal{D} \subset \mathbb{R}^2 \times [0,T]$ the domain of a spatiotemporal sequence, ∇ be the* spatial *gradient, $\nabla\cdot$ the* spatial *divergence on D and ν_s the spatial component of the exterior unit normal to the boundary $\partial\mathcal{D}$ of $\mathcal{D}$. Let $u : \mathcal{D} \to \mathbb{R}$ and $\vec{v} : \mathcal{D} \to \mathbb{R}^2$ be a $\mathcal{C}^1$-vector field on $\mathcal{D}$. Then*

$$\int_{\mathcal{D}} \nabla u \cdot \vec{v}\,dx = -\int_{\mathcal{D}} u\,\nabla\cdot\vec{v}\,dx + \int_{\partial\mathcal{D}} u\,\vec{v}\cdot\nu_s\,ds.$$

where s is the area measure on $\partial\mathcal{D}$.

Proposition 4.2 *Let $\mathcal{D} \subset \mathbb{R}^2 \times [0,T]$ the domain of a spatiotemporal sequence, φ, $\psi : \mathcal{D} \to \mathbb{R}^k$, $k \geq 1$, $\vec{v} : \mathcal{D} \to \mathbb{R}^2$ be a $\mathcal{C}^1$-vector field and $\vec{V}$ its "spatiotemporal" extension, as introduced in Section 3.1. Then one has*

$$\begin{aligned}\int_{\mathcal{D}} (\mathcal{L}_v \varphi) \cdot \psi \, dr &= -\int_{\mathcal{D}} \varphi \, \nabla_3 \cdot (\psi \otimes V) \, dr + \int_{\partial\mathcal{D}} (\varphi \cdot \psi)\, (V \cdot \nu)\, ds \\ &= -\int_{\mathcal{D}} \varphi \cdot [\mathcal{L}_{\vec{v}} \psi + \psi \, \nabla \cdot \vec{v}] \, dr + \int_{\partial\mathcal{D}} (\varphi \cdot \psi) \left(\vec{V} \cdot \nu\right) ds\end{aligned}$$

where ν is the exterior unit normal to the oriented boundary $\partial\mathcal{D}$ of $\mathcal{D}$ and s is the measure on $\partial\mathcal{D}$ and $- \cdot -$ is used to denote inner products both on $\mathbb{R}^k$ and $\mathbb{R}^2$ and $\psi \otimes V = \left(\psi_1 V^\top, \ldots, \psi_k V^\top\right)^\top$ and the divergence is applied componentwise, i.e., $\nabla \cdot_3 (\psi \otimes V) = (\nabla \cdot_3 (\psi_i V))_{i=1\ldots k}$.

The actual velocity field may present discontinuities, as well as functions φ and ψ. The simple formula above must then be replaced by a more complex one, as some terms must be considered as measures [6], but we will not consider it in this work, the problem becoming theoretically very difficult. A practical way to handle it will be provided by the nonlinearity of the terms considered.

In the following gradient computations, we will for legibility reasons, omit the different λ_i weights from the energy components.

4.2.2 Data term

If one wants to use the inpainting/denoising formulation, term $E_1(u; u_0)$ defined in formula (15) must be differentiated with respect to u and using (18), one finds immediately that

$$dE_{1u}\xi = \int_{\Omega^c} (u - u_{0i})\, \xi \, dr, \quad \nabla_u E_1 = \chi_{\Omega^c}(u - u_0). \tag{19}$$

When $u_0 = (u_{0i})_{i=1\ldots k}$ and $u = (u_i)_{i=1\ldots k}$ have values in $\mathbb{R}^k$, equations above are simply replaced by

$$\nabla_{u_i} E_1 = \chi_{\Omega^c}(u_i - u_{01}), \quad i = 1, \ldots, k.$$

4.2.3 Spatial regularity term

The derivation of the gradient of the spatial regulariry term E_2 given in formula (16) is rather classical. We recall it however as we are interested in the boundary terms that come with it. A straightforward computation using directional derivatives and Prop. 4.1 leads to

$$dE_{2u}\xi = -\int_{\mathcal{D}} \xi \nabla \cdot (A \nabla u) \, dr \tag{20a}$$

$$+ \int_{\partial\mathcal{D}} \xi A \, (\nabla u \cdot \nu_s) \, ds \tag{20b}$$

where we have set

$$A := A(u) := \phi'(|\nabla u|^2). \tag{21}$$

In the pure inpainting case, $\mathcal{D} = \Omega$, and we restrict to deformations ξ that have compact support in Ω, which means that the boundary term (20b) vanishes here and the gradient is given by

$$\nabla E_{2u} = -\chi_\Omega \nabla \cdot (A \nabla u)\,. \tag{22}$$

When u is vector-valued, by using squared-gradient magnitude as defined in formula (17), one obtains the k terms

$$\nabla_{u_i} E_2 = -\chi_\Omega \nabla \cdot (A \nabla u_i)\,, \quad i = 1, \ldots, k$$

which are coupled via their common diffusivity A.

4.2.4 Temporal prior / motion likelihood term

Terms E_3^i couple intensity and motion. We need to compute their gradients with respect to intensity and with respect to motion, and the novel part is really the intensity gradient. For that reason we present computations for E_3^1 and E_3^2 as well as the result for E_3^3. Here too, integration domains depend on the type of problem we are interested in, it will be $\mathcal{D} = \Omega$ for pure inpainting, and $\mathcal{D} = D$ for inpainting denoising. Gradients with respect to motion fields are provided without any computation and we won't detail their adaptation in the multiresolution setting, which is standard and properly explained in the references on optical flow that we provide.

The term E_3^1. We start with term E_3^1 as defined in Table 1: a straightforward calculation using formula (18) and Prop 4.2 gives

$$\begin{aligned} dE_{3u}^1\xi &= -\int_{\mathcal{D}} \left[\mathcal{L}_{\vec{v}}(B_1\mathcal{L}_{\vec{v}}u) + B_1\left(\mathcal{L}_{\vec{v}}u\right)\nabla\cdot\vec{v}\right]\xi\, dr && \text{(23a)}\\ &\quad + \int_{\partial\mathcal{D}} \xi B_1\mathcal{L}_{\vec{v}}u\left(\vec{V}\cdot\nu\right)\, ds && \text{(23b)} \end{aligned}$$

where we have set

$$B_1 := B_1(u,\vec{v}) := \phi'(|\mathcal{L}_{\vec{v}}u|^2). \tag{24}$$

Here too, in the pure inpainting case, ξ has compact support on Ω and the boundary integral (23b) vanishes, the sought gradient is

$$\begin{aligned} \nabla_u E_3^1 &= -\chi_\Omega\mathcal{L}_{\vec{v}}[B_1\mathcal{L}_{\vec{v}}u] \\ &\quad -\chi_\Omega B_1\left(\mathcal{L}_{\vec{v}}u\right)\nabla\cdot\vec{v}. \end{aligned} \tag{25}$$

When u is vector valued, B_1 couples the different channels via formula (17) and one obtains k terms:

$$\nabla_{u_i} E_3^1 = -\chi_\Omega\left(\mathcal{L}_{\vec{v}}[B_1\mathcal{L}_{\vec{v}}u_i] + B_1\left(\mathcal{L}_{\vec{v}}u_i\right)\nabla\cdot\vec{v}\right), \quad i = 1,\dots,k.$$

The term $\nabla_{\vec{v}} E_3^1$ has been computed by many authors, see for instance [50], and, for the general vector-valued setting, is given, with our notations, by the vectorial expression:

$$\nabla_{\vec{v}} E_3^1 = B_1\sum_{i=1}^{k}\left(\mathcal{L}_{\vec{v}}u_i\right)\nabla u_i$$

and is associated with vanishing Neumann Boundary conditions.

The term E_3^2. The additional gradient magnitude term in energy E_3^2 induces higher order terms in the differential. Once the directional derivative is computed, in order to transform it as a sum of integrals on the domain and a boundary intergral, we need to apply Prop. 4.2 and Prop. 4.1. A careful but nevertheless straightforward computation provides

$$\begin{aligned} dE_{3u}^2\xi &= -\int_{\mathcal{D}} \xi\left[\mathcal{L}_{\vec{v}}(B_2\mathcal{L}_{\vec{v}}u) + B_2\left(\mathcal{L}_{\vec{v}}u\right)\nabla\cdot\vec{v}\right]\, dr && \text{(26a)}\\ &\quad +\gamma\int_{\mathcal{D}} \xi\nabla\cdot\left[\mathcal{L}_{\vec{v}}(B_2\mathcal{L}_{\vec{v}}\nabla u) + B_2\left(\mathcal{L}_{\vec{v}}\nabla u\right)\nabla\cdot\vec{v}\right]\, dr && \text{(26b)}\\ &\quad -\gamma\int_{\partial\mathcal{D}} \xi\left[\mathcal{L}_{\vec{v}}(B_2\mathcal{L}_{\vec{v}}\nabla u) + B_2\left(\mathcal{L}_{\vec{v}}\nabla u\right)\nabla\cdot\vec{v}\right]\cdot\nu_s\, ds && \text{(26c)}\\ &\quad +\int_{\partial\mathcal{D}} B_2\left[\xi\mathcal{L}_{\vec{v}}u + \gamma\nabla\xi\cdot\mathcal{L}_{\vec{v}}\nabla u\right]\left(\vec{V}\cdot\nu\right) ds && \text{(26d)} \end{aligned}$$

where we have set

$$B_2 := B_2(u,\vec{v}) := \phi'\left(|\mathcal{L}_{\vec{v}}u|^2 + \gamma|\mathcal{L}_{\vec{v}}\nabla u|^2\right). \tag{27}$$

Once again, for the pure inpainting case, boundary integrals 26c and 26d vanish and the corresponding gradient is

$$\begin{aligned}\nabla_u E_3^2 &= -\chi_\Omega \left[\mathcal{L}_{\vec{v}}(B_2\mathcal{L}_{\vec{v}}u) - \gamma\nabla\cdot\left(\mathcal{L}_{\vec{v}}(B_2\mathcal{L}_{\vec{v}}\nabla u)\right)\right] \\ &\quad -\chi_\Omega \left[B_2\left(\mathcal{L}_{\vec{v}}u\right)\nabla\cdot\vec{v} - \gamma\nabla\cdot\left(B_2\left(\mathcal{L}_{\vec{v}}\nabla u\right)\nabla\cdot\vec{v}\right)\right]. \end{aligned} \tag{28}$$

In the vector valued case, each component is given by the formula above, replacing u by u_i, the components coupling provides from B_2.

The gradient with respect to the motion field $\vec{v}$ of this term is, in the general vector valued case, given by

$$\nabla_v E_3^2 = B_2 \sum_{i=1}^{k} \left[\left(\mathcal{L}_{\vec{v}}u_i\right)\nabla u_i + \mathcal{H}(u_i)\mathcal{L}_{\vec{v}}(\nabla u_i)\right].$$

The term E_3^3. Similar computations can be performed for the term E_3^3 and similar complex boundary integral terms appear in the computations. We set

$$B_3 := B_3(u,\vec{v}) := \phi'(|\mathcal{L}_{\vec{v}}\nabla u|^2) \tag{29}$$

and together with B_1 that was defined in formula (24), we get

$$dE_{3u}^3\xi = -\int_{\mathcal{D}} \xi\left[\mathcal{L}_{\vec{v}}(B_1\mathcal{L}_{\vec{v}}u) + B_1\left(\mathcal{L}_v u\right)\nabla\cdot v\right] dr \tag{30a}$$

$$+\gamma\int_{\mathcal{D}} \xi\nabla\cdot\left[\mathcal{L}_{\vec{v}}(B_3\mathcal{L}_{\vec{v}}\nabla u) + B_3\left(\mathcal{L}_{\vec{v}}\nabla u\right)\nabla\cdot\vec{v}\right] dr \tag{30b}$$

$$-\gamma\int_{\partial\mathcal{D}} \xi\left[\mathcal{L}_{\vec{v}}(B_3\mathcal{L}_{\vec{v}}\nabla u) + B_3\left(\mathcal{L}_{\vec{v}}\nabla u\right)\nabla\cdot v\right]\cdot\nu_s\, ds \tag{30c}$$

$$+\int_{\partial\mathcal{D}} \left[\xi B_1\mathcal{L}_{\vec{v}}u + \gamma B_3\nabla\xi\cdot\mathcal{L}_{\vec{v}}\nabla u\right](\vec{V}\cdot\nu)\, ds \tag{30d}$$

In the pure inpainting case, boundary integrals vanish because of conditions on ξ and the sought gradient is

$$\begin{aligned}\nabla E_{3u}^3 &= -\chi_\Omega \left[\mathcal{L}_{\vec{v}}(B_1\mathcal{L}_{\vec{v}}u) - \gamma\nabla\cdot\left(\mathcal{L}_{\vec{v}}(B_3\mathcal{L}_{\vec{v}}\nabla u)\right)\right] \\ &\quad -\chi_\Omega \left[B_1\left(\mathcal{L}_{\vec{v}}u\right)\nabla\cdot\vec{v} - \gamma\nabla\cdot\left(B_3\left(\mathcal{L}_{\vec{v}}\nabla u\right)\nabla\cdot\vec{v}\right)\right]. \end{aligned} \tag{31}$$

4.2.5 Boundary terms for minimization with respect to image

So far we have not discussed boundary conditions for the inpainting/denoising problem where the integration domain is D and we cannot assume that a deformation direction vanishes along ∂D. The resulting boundary integrals are rather complex. We consider the case arising from combining spatial regularity term E_2 and temporal regularity term E_3^1. The resulting boundary integral comes from (20b) and (23b):

$$\int_{\partial D} \xi\left[\lambda_1\nabla u\cdot\nu_s + \lambda_3 B_1\mathcal{L}_{\vec{v}}u(\vec{V}\cdot\nu)\right] ds$$

and the natural boundary condition is thus to impose that

$$\lambda_1\nabla u\cdot\nu_s + \lambda_3 B_1\mathcal{L}_{\vec{v}}u(\vec{V}\cdot\nu) = 0.$$

On domain D exterior normals are respectively $\nu = \pm(1,0,0)^T$, $\pm(0,1,0)^T$ and $\pm(0,0,1)^T$ corresponding to "vertical", "horizontal" and "temporal" faces of the domain. With these vectors, the conditions become

$$u_x = -\lambda_3 v_1 B_1 \frac{u_y v_2 + u_t}{\lambda_2 + \lambda_3 v_1 B_1}, \quad \text{resp. } u_y = -\lambda_3 v_y B_1 \frac{u_x v_1 + u_t}{\lambda_2 + \lambda_3 v_2 B_1}, \quad \text{resp. } u_t = -(u_x v_1 + u_y v_2)$$

if one uses the expansion $\mathcal{L}_{\vec{v}}u = u_x v_1 + u_y v_2 + u_t$. However, such an expansion is problematic numerically. Indeed, it is well known in the optical flow literature that such an expansion requires sufficient smoothness

on u and/or small motion, and when these requirements are not fulfilled, one has to use approximations such as the one given in formulas (5) and (6) in the previous section.

A stronger requirement can be imposed, that both $\mathcal{L}_{\vec{v}}u$ and $\nabla u \cdot \nu_s$ vanish simultaneously. If it holds, then the boundary integral vanishes, but imposing such a requirement may lead to an overdetermined system: for instance, on the "vertical" face, using the above Lie derivative expansion, it becomes

$$\begin{cases} u_x = 0 \\ u_y v_2 + u_t = 0 \end{cases}$$

and while the first equation expresses the absence of variation across the boundary, the second is problematic as u_y, u_t and v_2 are generally estimated from $\vec{v}$ and u from already known values of the image.

We will nevertheless assume that $\mathcal{L}_{\vec{v}}u = \nabla u \cdot \nu_s = 0$ as, the use of standard schemes for computing gradient and the schemes that come from (5) and (6) for computing $\mathcal{L}_{\vec{v}}u$ provide a simple and efficient *numerical* treatment of these boundary conditions. These can also be extended to the terms ${E_3}^2$ and E_3^3 by requiring also that $\mathcal{L}_{\vec{v}}\nabla u = 0$ on ∂D. In fact, without such an assumption, one cannot get rid of boundary terms appearing in the computations of differentials for ${E_3}^2$ and E_3^3, where not only a deformation ξ appears, but also its spatial gradient $\nabla\xi$. What makes the difference with the pure inpainting formulation is that it appears extremely difficult to control what happens at scene boundary. If one could assume null camera motion and moving objects remaining in the image domain, boundary difficulties would vanish, but this is in general a much too severe restriction.

What usually counterbalances the inconsistancies of our choice of boundary conditions is the reaction to the data term: we are forced to stay reasonably close to the observed data u_0, along the boundary too.

With these assumptions, the three gradient terms $\nabla_u E_3^1$, $\nabla_u E_3^2$, and $\nabla_u E_3^3$ are easily computed in the inpainting-denoising formulation and are given by

$$\begin{aligned} \nabla_u E_3^1 &= -\mathcal{L}_{\vec{v}}[B_1 \mathcal{L}_{\vec{v}}u] \\ &\quad -B_1\left(\mathcal{L}_{\vec{v}}u\right) Div\, \vec{v}, \end{aligned}$$

$$\begin{aligned} \nabla_u E_3^2 &= -\left[\mathcal{L}_{\vec{v}}(B_2 \mathcal{L}_{\vec{v}}u) - \gamma\nabla\cdot\left(\mathcal{L}_{\vec{v}}(B_2 \mathcal{L}_{\vec{v}}\nabla u)\right)\right] \\ &\quad -\left[B_2\left(\mathcal{L}_{\vec{v}}u\right)\nabla\cdot\vec{v} - \gamma\nabla\cdot\left(B_2\left(\mathcal{L}_{\vec{v}}\nabla u\right)\nabla\cdot\vec{v}\right)\right], \end{aligned}$$

$$\begin{aligned} \nabla E_{3u}^3 &= -\left[\mathcal{L}_{\vec{v}}(B_1 \mathcal{L}_{\vec{v}}u) - \gamma\nabla\cdot\left(\mathcal{L}_{\vec{v}}(B_3 \mathcal{L}_{\vec{v}}\nabla u)\right)\right] \\ &\quad -\left[B_1\left(\mathcal{L}_{\vec{v}}u\right)\nabla\cdot\vec{v} - \gamma\nabla\cdot\left(B_3\left(\mathcal{L}_{\vec{v}}\nabla u\right)\nabla\cdot\vec{v}\right)\right]. \end{aligned}$$

Two important remarks regarding the structure of these terms (also valid for their pure inpainting counterpart):

- First, one observes that these three gradient terms $\nabla_u E_3^1$, $\nabla_u E_3^2$, and $\nabla_u E_3^3$ are each decomposed in two parts, a part containing a double differentiation along the flow field via $\mathcal{L}_{\vec{v}}()$, which corresponds to diffusion along a flow line, and a term where $\nabla\cdot\vec{v}$ appears, which corrects for the non parallelism of the motion field $\vec{v}$. These terms are non linear transport along trajectories of the velocity field and are controled by the flow divergence. This means that not only the punctual value of $\vec{v}$ must be taken into account but also its variations. This is a consequence of Prop. 4.2 and is to be put in parallel with duality ideas of Florack *et al.* in [24], but see also [52, 8, 39].

- The second remark concerns the assumptions made on conserved quantities in order to build the different terms E_3^i. The corresponding image gradients make clear that these conservations should be enforced for the image minimizers of these energies, with intensity diffusion for intensity conservation and spatial gradient diffusion for spatial gradient conservation.

4.2.6 Flow regularity terms

As mentioned already, their regularizers E_4^i, $i = 1, 2, 3$, are borrowed from existing motion recovery algorithms, and their gradient calculations, with its multiresolution adaptation, have been presented in the

papers already mentioned. We just briefly recall the results.

The term E_4^1. The two gradient components are independent, and if one sets

$$C_1^1 := C_1^1(\vec{v}) := \phi'\left(|\nabla v_1|^2\right), \quad C_1^2 := C_1^2(\vec{v}) := \phi'\left(|\nabla v_2|^2\right),$$

one gets

$$\nabla_{\vec{v}} E_4^1 = -\begin{pmatrix} \nabla\cdot\left(C_1^1 \nabla v_1\right) \\ \nabla\cdot\left(C_1^2 \nabla v_2\right) \end{pmatrix}.$$

The term E_4^2. The two gradient components are coupled via the common diffusivity

$$C_2(\vec{v}) = \phi'\left(|\nabla v_1|^2 + |\nabla v_2|^2\right)$$

and

$$\nabla_{\vec{v}} E_4^2 = -\begin{pmatrix} \nabla\cdot\left(C_2 \nabla v_1\right) \\ \nabla\cdot\left(C_2 \nabla v_2\right) \end{pmatrix}.$$

The term E_4^3. This term is a simple spatio-temporal extension of the previous one, the new diffusivity is

$$C_3(\vec{v}) = \phi'\left(|\nabla_3 v_1|^2 + |\nabla_3 v_2|^2\right)$$

and

$$\nabla_{\vec{v}} E_4^3 = -\begin{pmatrix} \nabla\cdot\left(C_3 \nabla_3 v_1\right) \\ \nabla\cdot\left(C_3 \nabla_3 v_2\right) \end{pmatrix}$$

where the divergences are spatio-temporal here since we use the spatio-temporal gradients of the v_i.

We note that these gradients formulations are naturally associated to Neumann boundary conditions on the components of $\vec{v}$, the main derivation mechanism being used is given by Prop. 4.1.

5 Discretization

For optical flow recovery parts, we use discretizations that have been proposed in the papers where we "borrowed" this terms. The discretization of spatial (and spatio-temporal) divergence terms are quite standard in the literature, see for instance [16, 6] and won't be discussed in this paper.

We will thus exclusively concentrate on the part of the inpainting equations that come from diffusion along flow lines i.e., terms of the type E_3^i. Because, in the numerics we need to use time discretized sequences, displacement fields must be used instead of velocity fields, we modify in a natural way our formulations to handle them, with classical warping techniques. These displacement fields do usually provide subpixel accuracy, making necessary the use of some form of spatial interpolation, which in turn make cumbersome the direct development of completely discrete schemes. In order to avoid that, we use an intermediary formulation where only the time part is discretized.

We will first study the semi-discrete schemes, then discuss the discretization of spatial terms of the form $\nabla\cdot\left(a\nabla u\right)$ that appear both in the spatial regularizer term and the tow higher order $\nabla_u E_3^i$, $i = 2, 3$. the spatial discretization. In the inpainting denoising settings, we need to discretize the gradient of the term E_1 as given by (19), this almost trivial, once one has taken care of multiresolution discretization for χ_Ω. We thus discuss briefly points related to the multiresolution setting. We end by discussing the approaches we have used to implement the final numerical solutions of the algebraic equations obtained from discretization.

In the sequel we will assume that we have a spatio-temporal grid with spatial grid spacing h_s (we assume the spatial grid to be squared, for simplicity) and temporal grid spacing h_t. Given a function $f : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}^k$, its temporal discretization will give rise to a family

$$(f_k)_{k\in\mathbb{Z}}, \quad f_k = f(-, -, kh_t) : \mathbb{R}^2 \to \mathbb{R}^k.$$

We will not consider explicitly the full discretization, as the general need to estimate several quantities at non-grid point locations would make it very cumbersome.

5.1 A time-discrete scheme.

In order to discretize the terms $\nabla_{\vec{v}} E_3^i$, we start with a temporal-only discretization of them, and this for two main reasons. First, displacement fields have generally subpixel accuracy, necessitating some spatial interpolation. Second: temporal discrete sequence modeling is at the heart of multiresolution / warping handling of motion recovery and most of these warping based motion recovery methods all use at some point the Lie Derivative semi-discrete approximation in formula (5).

We will indicate some of the differences that naturally appear in the gradient terms with respect to flow in this setting.

In order to develop schemes, we start from the continuous energy terms that we discretize in time and derive analogues of Prop 4.2, using the Lie derivative forward difference expression defined in formula (5). To get rid of the boundary conditions difficulties that arise from boundary integral in Prop 4.2, they would be much worse in the semi-discrete case, we extend artificially the spatial domain to the whole of $\mathbb{R}^2$ and the temporal range to the whole of $\mathbb{R}$, while we will assume that both the sequence and the velocity field have compact support.

Let $\phi, \psi : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}^k$ be sequence, $(\phi_k)_{k\in\mathbb{Z}}$ and $(\psi_k)_{k\in\mathbb{Z}}$ their temporal discretizations. Denote by $\mathbf{v}_k : \mathbb{R}^2 \to \mathbb{R}^2$ the displacement field $\mathbf{v}(-, kh_t, h_t)$ where $\mathbf{v}(\mathbf{x}, t, h)$ was defined in formula (7). Following (5) and (6), we set, with some abuse of notation, and dropping the subscript t from the temporal spacing h_t,

$$\left(\mathcal{L}_{\mathbf{v}}^{h+}\phi\right)_k := \frac{\phi_{k+1}(\mathbf{x}+\mathbf{v}_k(\mathbf{x})) - \phi_k(\mathbf{x})}{h}, \quad \left(\mathcal{L}_{\mathbf{v}}^{h-}\phi\right)_k := \frac{\phi_k(\mathbf{x}) - \phi_{k-1}(\mathbf{x}-\mathbf{v}_k(\mathbf{x}))}{h}, \quad k \in \mathbb{Z}.$$

For $k \in \mathbb{Z}$, set $H_k = I + \mathbf{v}_k$, where I is the identity map of $\mathbb{R}^2$. For h small enough, this is a local diffeomorphism if the velocity field v is smooth. Because of the compact support assumption, v is bounded in norm, implying that for h small enough, H_k is a diffeomorphism.

Proposition 5.1 *Assume that the H_k's are global diffeomorphisms. Define $(K_k)_k$ by $K_{k+1} = H_k^{-1}$, and set $\mathbf{w}_k = I - K_k$. If $|JK_k|$ is the Jacobian determinant of K_k, $|JK_k| = 1 - \nabla\cdot\mathbf{w}_k + |J\mathbf{w}_k|$ and the following holds:*

$$\sum_{k\in\mathbb{Z}} \int_{\mathbb{R}^2} \left(\mathcal{L}_{\mathbf{v}}^{h+}\phi\right)_k(\mathbf{x})\psi_k(\mathbf{x})\,d\mathbf{x} = -\sum_{k\in\mathbb{Z}} \int_{\mathbb{R}^2} \phi_k \left(\left(\mathcal{L}_{\mathbf{w}}^{h-}\psi\right)_k - \frac{\nabla\cdot\mathbf{w}_k - |J\mathbf{w}_k|}{h}\psi_{k-1}\circ K_k \right) d\mathbf{x}.$$

The proof is straightforward via the change of variables theorem and renaming of summation and integration variables. The *backward displacement field* $\mathbf{w}_k$ is given by

$$\mathbf{w}_k(\mathbf{x}) = -\int_0^h \vec{v}(\mathbf{x}(kh-\tau), kh-\tau)\,d\tau.$$

and passing to the limit $h \to 0$, $kh \to t_0$ in the above proposition, one easily gets Prop. 4.2 (it is easy to check that in the limit $\nabla\cdot\mathbf{w}_k/h \to -\nabla\cdot\vec{v}$ and $|J\mathbf{w}_k|/h \to 0$) with the same problems in term of modeling: $\vec{v}$ is generally not smooth: it should present some discontinuities, due to occlusion/disocclusion. Here too, theoretical and practical difficulties appear, and we do not consider this case, a practical solution is partially provided by the use of non quadratic data and regularizing terms in the variational formulations, and some smoothness can anyway be justified by the smoothing effect of the discretization.

Note that the result stated in Prop. 5.1 is essentially an adjunction result for the Hilbert space of functions families with inner product

$$(\boldsymbol{\phi}|\boldsymbol{\psi}) = \sum_{k\in\mathbb{Z}} \int_{\mathbb{R}^2} \phi_k\psi_k\,d\mathbf{x}$$

with the associated notion of gradient that we use in the sequel.

We will use the displacement fields and warp notations from Prop. 5.1 and as we had introduced in Subsection 4.2.4, the spatio-temporal coefficient functions B_i, $i = 1, 2, 3$, we introduce their counterparts sequences $B_i^{\pm}$, $i = 1, 2, 3$, respectively defined by:

$$B_{1k}^{+} = \phi'\left(\left(\mathcal{L}_{\mathbf{v}}^{h+}u\right)_k^2\right), \quad B_{1k}^{-} = \phi'\left(\left(\mathcal{L}_{\mathbf{w}}^{h-}u\right)_k^2\right),$$

$$B_{2k}^{+}=\phi'\left(\left(\mathcal{L}_{\mathbf{v}}^{h+}u\right)_k^2+\gamma\left|\mathcal{L}_{\mathbf{v}}^{h+}\nabla u\right|_k^2\right),\quad B_{2k}^{-}=\phi'\left(\left(\mathcal{L}_{\mathbf{w}}^{h-}u\right)_k^2+\gamma\left|\mathcal{L}_{\mathbf{w}}^{h-}\nabla u\right|_k^2\right),\quad\text{and}$$
$$B_{3k}^{+}=\phi'\left(\left|\mathcal{L}_{\mathbf{v}}^{h+}\nabla u\right|_k^2\right),\quad B_{3k}^{-}=\phi'\left(\left|\mathcal{L}_{\mathbf{w}}^{h-}\nabla u\right|_k^2\right).$$

We will also set

$$F_k=\frac{\nabla\cdot\mathbf{w}_k-|J\mathbf{w}_k|}{h}. \tag{32}$$

We look at the simplest term E_3^1 and detail the derivation of its semi-discrete gradient. To be in the assumptions of the above proposition, we extend the spatial domain to the whole of $\mathbb{R}^2$ and assume that the both the image sequence and velocity fields have compact support. Let $\mathbf{u}=(u_k)_k$ the family of image frames. Using forward difference approximation of Lie derivative, the semi-discretization of E_3^1 is

$$\bar{E}_3^1(\mathbf{u},\mathbf{v})=\frac{1}{2}\sum_{k\in\mathbb{Z}}\int_{\mathbb{R}^2}\phi\left(\left(\mathcal{L}_{\mathbf{v}}^{h+}u\right)_k^2\right)\,d\mathbf{x}.$$

In order to compute its gradient with respect to $\mathbf{u}$, we use the directional derivative approach, and for $\boldsymbol{\xi}=(\xi_k)_k$, we compute $d\bar{E}_{\mathbf{u}3}^1\boldsymbol{\xi}=\ell'(0)$ where $\ell(\tau)=\bar{E}_3^1(\mathbf{u}+\tau\boldsymbol{\xi},\mathbf{v})$ to obtain

$$d\bar{E}_{3\mathbf{u}}^1\boldsymbol{\xi}=\sum_{k\in\mathbb{Z}}\int_{\mathbb{R}^2}B_{1k}^{+}\left(\mathcal{L}_{\mathbf{v}}^{h+}u\right)_k\left(\mathcal{L}_{\mathbf{v}}^{h+}\xi\right)_k\,d\mathbf{x}.$$

Applying the above proposition, one gets

$$d\bar{E}_{3\mathbf{u}}^1\boldsymbol{\xi}=-\sum_{k\in\mathbb{Z}}\int_{\mathbb{R}^2}\xi_k\left[\mathcal{L}_{\mathbf{w}}^{h-}\left(B_1^{+}\left(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u}\right)\right)_k-F_k\left(\left(B_{1k-1}^{+}\left(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u}\right)_{k-1}\right)\circ K_k\right)\right]\,d\mathbf{x}$$

from which the k-component of the sought gradient can be written as

$$\left(\nabla_{\mathbf{u}}\bar{E}_1^2\right)_k=-\mathcal{L}_{\mathbf{w}}^{h-}\left(B_1^{+}\left(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u}\right)\right)_k+F_k\left(\left(B_{1k-1}^{+}\left(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u}\right)_{k-1}\right)\circ K_k\right)$$

But, by a straightforward calculation, since $K_k=H_{k-1}^{-1}$, one gets that

$$\left(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u}\right)_{k-1}\circ K_k=\left(\mathcal{L}_{\mathbf{w}}^{h-}\mathbf{u}\right)_k,\quad B_{ik-1}^{+}\circ K_k=B_{ik}^{-}.$$

For a given $\mathbf{x}\in\mathbb{R}^2$, set $\mathbf{x}_k^{+}=H_k(\mathbf{x})=\mathbf{x}+\mathbf{v}_k(\mathbf{x})$, $\mathbf{x}_k^{-}=K_k(\mathbf{x})=\mathbf{x}-\mathbf{w}_k(\mathbf{x})$. Then we can write

$$\left(\nabla_{\mathbf{u}}\bar{E}_3^1\right)_k(\mathbf{x}) = -\frac{B_{1k}^{-}u_{k-1}(\mathbf{x}_k^{-})-(B_{1k}^{-}+B_{1k}^{+})u_k(\mathbf{x})+B_{1k}^{+}u_{k+1}(\mathbf{x}_k^{+})}{h^2} \tag{33a}$$
$$+ B_{1k}^{-}\frac{u_k(\mathbf{x})-u_{k-1}(\mathbf{x}_k^{-})}{h}F_k. \tag{33b}$$

The spatial grid of the problem is usually imposed, and a spatial interpolation technique is necessary to compute the $u(\mathbf{x}_k^{\pm})$ and the $B_i^{\pm}$. In this work, we have used bilinear and bicubic interpolations. A natural question, when implementing schemes based on the above partial discretization is what to do when $\mathbf{x}_k^{\pm}$ falls out of the numerical domain. The easiest solution is then use the value at $\mathbf{x}$ instead, this means that we impose $(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u})_k=0$ (resp. $(\mathcal{L}_{\mathbf{w}}^{h-}\mathbf{u})_k=0$) which are the numerical translations of the boundary conditions we discussed in Paragraph 4.2.5. The same type of operations are applied for the semi-discrete version of energy E_3^2

$$\bar{E}_3^2(\mathbf{u},\mathbf{v})=\frac{1}{2}\sum_{k\in\mathbb{Z}}\int_{\mathbb{R}^2}\phi\left(\left(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u}\right)_k^2+\gamma\left|\left(\mathcal{L}_{\mathbf{v}}^{h+}\nabla\mathbf{u}\right)_k\right|^2\right)\,d\mathbf{x}.$$

This lead to the following gradient

$$\begin{aligned}
\left(\nabla_u \bar{E}_3^2\right)_k &= -\frac{B_{2k}^- u_{k-1}(\mathbf{x}_k^-) - (B_{2k}^- + B_{2k}^+)u_k(\mathbf{x}) + B_{2k}^+ u_{k+1}(\mathbf{x}_k^+)}{h^2} && \text{(34a)}\\
&+ B_{2k}^- \frac{u_k(\mathbf{x}) - u_{k-1}(\mathbf{x}_k^-)}{h} F_k && \text{(34b)}\\
&+ \gamma\nabla\cdot\left(\frac{B_{2k}^- \nabla u_{k-1}(\mathbf{x}_k^-) - (B_{2k}^- + B_{2k}^+)\nabla u_k(\mathbf{x}) + B_{2k}^+ \nabla u_{k+1}(\mathbf{x}_k^+)}{h^2}\right) && \text{(34c)}\\
&- \gamma\nabla\cdot\left(B_{2k}^- \frac{\nabla u_k(\mathbf{x}) - \nabla u_{k-1}(\mathbf{x}_k^-)}{h} F_k\right) && \text{(34d)}
\end{aligned}$$

while, for energy

$$\bar{E}_3^3(\mathbf{u},\mathbf{v}) = \frac{1}{2}\sum_{k\in\mathbb{Z}}\int_{\mathbb{R}^2}\left[\phi\left(\left(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u}\right)_k^2\right) + \gamma\phi\left(\left|\left(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u}\right)_k\right|^2\right)\right]\,d\mathbf{x}$$

the gradient is given by

$$\begin{aligned}
\left(\nabla_u \bar{E}_3^3\right)_k &= -\frac{B_{1k}^- u_{k-1}(\mathbf{x}_k^-) - (B_{1k}^- + B_{1k}^+)u_k(\mathbf{x}) + B_{1k}^+ u_{k+1}(\mathbf{x}_k^+)}{h^2} && \text{(35a)}\\
&+ B_{1k}^- \frac{u_k(\mathbf{x}) - u_{k-1}(\mathbf{x}_k^-)}{h} F_k && \text{(35b)}\\
&+ \gamma\nabla\cdot\left(\frac{B_{3k}^- \nabla u_{k-1}(\mathbf{x}_k^-) - (B_{3k}^- + B_{3k}^+)\nabla u_k(\mathbf{x}) + B_{3k}^+ \nabla u_{k+1}(\mathbf{x}_k^+)}{h^2}\right) && \text{(35c)}\\
&- \gamma\nabla\cdot\left(B_{3k}^- \frac{\nabla u_k(\mathbf{x}) - \nabla u_{k-1}(\mathbf{x}_k^-)}{h} F_k\right). && \text{(35d)}
\end{aligned}$$

In the three cases care must be taken when dealing with the transport-like terms that appear in the different expressions above. While schemes resulting from (33) are relatively easy to implement, modulo of course transport-like term, care is to be taken for schemes (34) and (35) due to the presence of spatial divergence. When properly dealt with, a numerical scheme is then available.

Before continuing, important points regarding the above energies and schemes must be addressed.

- The use of the forward difference approximations in the energies have also an effect on their gradients with respect to the displacement field variable. An elementary computation for the case of $\bar{E}_3^1$ gives

  $$\left(\nabla_{\mathbf{v}} \bar{E}_3^1\right)_k = B_1 k^+ \left(\mathcal{L}_{\mathbf{v}}^{h+}\mathbf{u}\right)_k \nabla u_{k+1} \circ H_k$$

  where the warping H_k appears naturally and corresponds precisely to the type of computations performed in multiresolution motion recovery. Remark that only H_k is used there and that there is no assumption on its inversibility.

- Being coherent with Lie derivatives approximations above, our motion recovery algorithm will normally return only the forward displacement field $\mathbf{v} = (\mathbf{v}_k)_k$ of the sequence, and thus we have only directly access to the forward warp H_k. How to compute the backward displacement field $\mathbf{w}$? Inversion of the H_k is generally not an option: even in the case where H_k was a true diffeomorphism, this would extremely complex. Moreover occlusion and disocclusion phenomena make it impossible. However, a simple solution consists in computing this backward flow from the image sequence itself. This should provide a reasonable solution for at least a good reason: Given an image $u : \mathbb{R}^2 \times [0,T] \to \mathbb{R}$, let us denote by $\hat{u}$ the "time-reversed" sequence obtained as

  $$\hat{u}(\mathbf{x},t) := u(\mathbf{x}, T-t).$$

  Assume that u is *smooth*. Then, taking the flow related part of our energy, i.e. $E_{ij} = \lambda_3 E_3^i + \lambda_4 E_4^j$, an elementary computation shows that if v minimizes E_{ij}, so is $-v$ for the image sequence $\hat{u}$. We could replace the optical flow estimation by a symmetrized version, in the spirit of the work of Alvarez *et al.* in [1], but this would deeply modify the formulations above.

- All the above gradients have some transport-like terms, e.g the term (33b) in the expression of $\nabla_{\mathbf{u}}\bar{E}_3^1$. Such a term may be difficult to handle numerically. One can ask whether it is necessary. In an informal way, we do expect that the diffusion term (33a) alone will smooth variations of u along the flow lines, and thus decrease the energy, i.e. that even when forgetting the transport term, the resulting expression would still be a descent direction for the energy, so should be consider for both explicit gradient descent resolution as well as some relaxation schemes. This however no so simple, as (33a) involves not only forward Lie derivatives but also backward ones while only forward ones are present in the semi-discrete energy. In the other hand, our flow regularizers generally favor displacement fields with small divergence, thus generally reducing the influence of the transport part.

5.2 Multiresolution details

As mentioned in Section 4.1, we solve the equations in a multiresolution framework. A few points should be mentioned. When building the multiresolution pyramid, image coarsening is necessary, and is in fact already handled by the motion estimation solver. What is not handled is the coarsening of the inpainting mask, i.e the numeric characteristic function of the missing data locus Ω. Coarsening of the image is usually performed by smoothing ans subsampling or any kind of method that has a proper low pass property. Coarsening of χ_Ω may be problematic when the same method is used. It blurs the boundary of Ω, and while this blurring may seem coherent, as it may indicate that a given pixel at coarse resolution contains some known partial information, we have constated that it often slows convergence down. We have used instead the somehow rough but simple nearest neighbor approach that guaranties that the mask remains binary valued and not become "too large" when coarsening.

In the other hand, one also need to interpolate the inpainting result from a given level to the next finer level, while interpolation of the motion is performed already by the flow recovery algorithm. In this work, image interpolation has been performed using a simple bilinear interpolation for inpainting denoising, while for pure inpainting, values obtained by bilinear interpolation and not in the missing data locus have been replaced by original image values downsampled at that resolution.

5.3 Solving the equations

Inpainting equations involving terms E_3^2 and E_3^3 are 4th-orders partial differential equations, with diffusion of gradients. This type of higher order diffusion may not obey the minimum-maximum principle, see for instance [27]. We rely therefore on a gradient descent scheme in these cases. For each resolution level we do the following. Having chosen an evolution step $d\tau$, we create the family $\mathbf{u}^n = (u_k^n)_k$, $n \geq 0$ and write a standard Eulerian step

$$\frac{u_k^{n+1}(\mathbf{x}) - u_k^n(\mathbf{x})}{d\tau} = -\left(\nabla_{\mathbf{u}^n}\bar{E}(\mathbf{x})\right)_k$$

where $\bar{E}$ is either of the form $\lambda_1\bar{E}_1+\lambda_2\bar{E}_2+\lambda_3E_3^i$ in the inpainting-denoising case, with $(\mathbf{x}, k)$ running over the full spatio-temporal grid in that case, while in the pure inpainting case $\bar{E}$ has the form $\lambda_2\bar{E}_2 + \lambda_3E_3^i$ and $(\mathbf{x}, k)$ runs only on the missing data locus, i.e the discretized and downsampled copy of Ω.

In the implementation, we choose once for all the evolution step $d\tau$ as well as the number N of evolution steps we perform. As in most explicit schemes, $d\tau$ must be chosen small enough, and this has a drastic impact on the running-time, particularly in the case of inpainting denoising.

In the case where we use E_3^1, we may consider, in a fixed point approach, linearizing the resulting equations and solvers such as Gauss-Seidel could be used. In order to be able to compare the different methods, we have however in that work only used the explicit Gradient Descent approach.

6 Experimental Evaluation

We present results for several of the algorithms we have discussed, on synthetic and real sequences. We follow the nomenclature of energy terms presented in Section 3, Tables 1 and 2. To accommodate the variety of intensity ranges in the different sequences, intensities have been linearly normalized to range

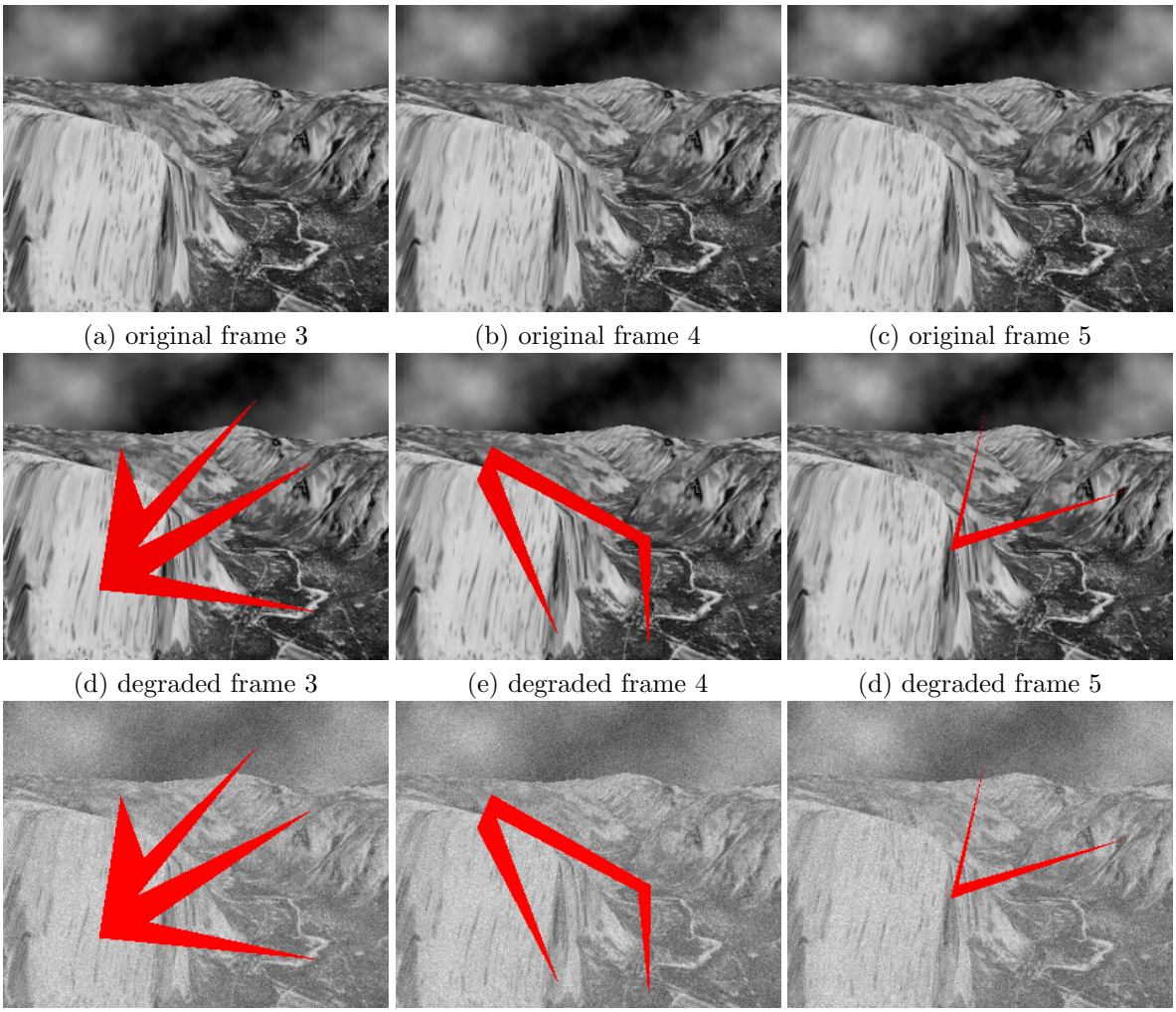
$[0, 1]$. This imply that numerical partial derivatives are bounded in absolute value, by bounds of the same order of magnitude, which in turns influences the practical range of ε values in $\phi(x^2) = \sqrt{x^2 + \varepsilon^2}$, we have taken $\varepsilon = 10^{-3}$. In all gradient descents, we have used an algorithmic time step $dt = 10^{-3}$. This in turns influences the choices of the different weights λ_i for the corresponding energies.

Although flow regularization term E_4^1 has been used, we only report results for term E_4^2 in the experiments we present here, as it has been argued [11, 12] that temporal regularity prior generally improves motion estimation.

We present a series of stills in this section, the reader should also look at the companion sequences, they are available at the location `http://image.diku.dk/francois/seqinp`. Some previous versions were available at the the companion web site `http://www-sop.inria.fr/books/imath` for the monograph of Aubert and Kornprobst [6].

The first one is the well known `Yosemite` sequence, created by Lynn Quam, very often used for optical flow evaluation. The sequence is artificial and the ground truth is known for the flow. It was degraded by removing large polygonal patches, three of them *overlapping consecutively* in time, on 6 of the 15 frames.

Figure 2 shows frames 2, 3, and 4 of the original sequence, corresponding degraded frames, and the noisy degraded ones where Gaussian noise of standard deviation 5% of the intensity range was added.



(a) original frame 3 (b) original frame 4 (c) original frame 5

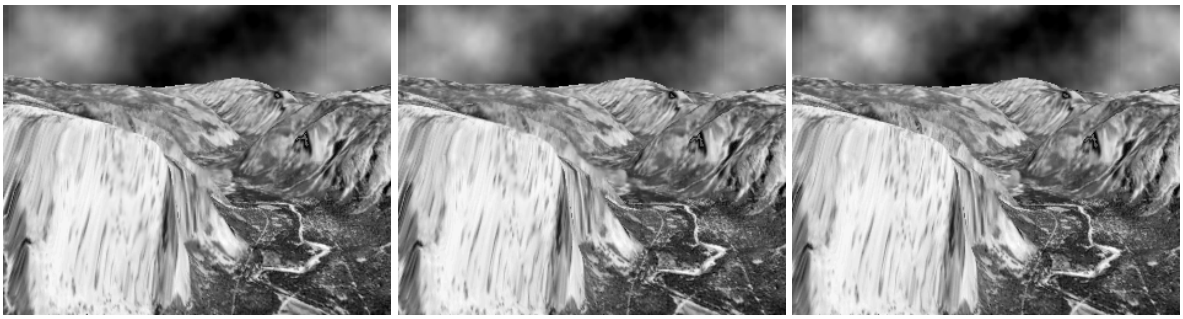
(d) degraded frame 3 (e) degraded frame 4 (d) degraded frame 5

Figure 2: `Yosemite` sequence. Original and degraded frames. The degradation in frame 3 is very large and one can notice that the 3 holes overlap around the frame centers.

We first run a simplified experiment where only the pure inpainting equation derived from terms E_2 and E_3^1 is solved and then the inpainting/denoising using the same spatial and trajectory smoothness

terms. We use the ground truth forward flow $\vec{v}_f$. We deal with the absence of ground truth for the backward optical flow $\vec{v}_b$ by computing it from the optical flow PDE derived from terms E_3^1 and E_4^2, having reversed time in the sequence and using as starting guess $t \mapsto -\vec{v}_f(T-t)$ where T is the forward flow sequence last time/frame.

Figure 3 present the results of these two experiments on the three degraded frames show in Fig. 2. For pure Inpainting, parameters where $\lambda_2 = 0.1$ (spatial regularization weight), $\lambda_3 = 1$ (flow lines regularization weight). For inpainting / denoising, the data weight λ_1 has been set to 20, while $\lambda_2 = \lambda 3 = 1$. Results of pure inpainting presented in (a), (b) and (c) are generally very good, one may however notice a lack on sharpness in the center of frames (a) and (b). Unsurprisingly, inpainted/denoised results present characteristics of Total Variation regularized images, some low scale details have been lost. Boulanger *et al.* [10] as well as Buades *et al.* [14] have shown that image sequence denoising is best achieved with patch based / non local means methods. Nevertheless, they cannot cope as is with large missing data. It may be worth investigating developing an hybrid method in that case.



(a) pure inpainting frame 3 (b) pure inpainting frame 4 (c) pure inpainting frame 4

(d) inpainted/denoised frame 3 (e) inpainted/denoised frame 4 (f) inpainted/denoised frame 5

Figure 3: Pure Inpainting and Inpainting/Denoising based on ground truth motion. While pure inpainted sequence is of excellent quality, the inpainted/denoised shows the usual patterns of Total-Variation like regularization with loss of fine scale details.

In the second set of experiments, we abandon the ground truth and run full recovery algorithms on the `Yosemite` sequence. In the first one, we run the pure inpainting algorithm corresponding to energy $\lambda_2 E_2 + \lambda_3 E_3^1 + \lambda_4 E_4^2$ with $\lambda_2 = 0.1$, $\lambda_3 = 1$ and $\lambda_4 = 0.2$, four pyramid levels where the number of pixels is roughly divided by 2 from one level to the next coarser one. Then Energy $\lambda_2 E_2 + \lambda_3 E_3^1 + \lambda_4 E_4^2$ has been used, withe the gradient weight $\gamma = 0.1$, and *these parameters, as well as pyramid sizes, have been used in all the remaining experiments presented in this work, except for the* ***Manon*** *sequence.* Results are shown in Figure 4. Spotting visually differences between the two sequences is difficult, although computing the differences, as illustrated on the last row of Figure 4 shows some, covering about 7% of the intensity range, while a plot of histograms of gradient magnitudes for the two results seems to indicate that this difference is in fact hardly significative. These histograms are shown in figure 5.

The second example is taken from the companion CD ROM of Kokaram's book [33]. We start with the `Mobile and Calendar` sequence, that have been extensively used for MPEG coding, deinterlacing and so on. It is a real one with 25 frames and with complex motion patterns. It is then artificially degraded to simulate blotches. (approximately 6% of the image is degraded with blocthes of multiple size, they may

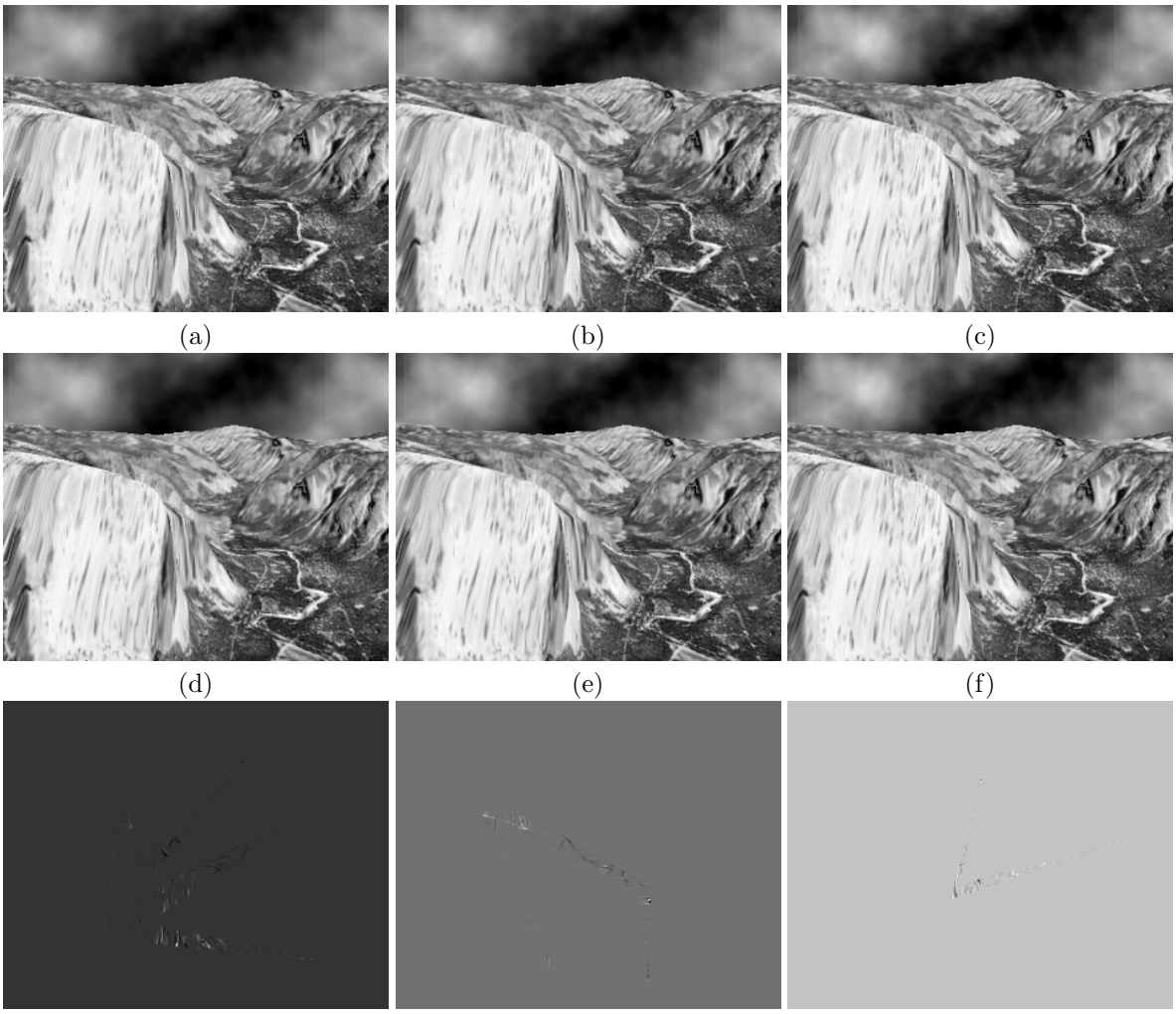


Figure 4: Full Image and Motion recovery in the case of pure Inpainting. The first row from minimization of low-order energy $E_2 + E_3^1 + E_4^2$, the second from higher order energy $E_2 + E_3^2 + E_4^2$ and the last row shows the differences.

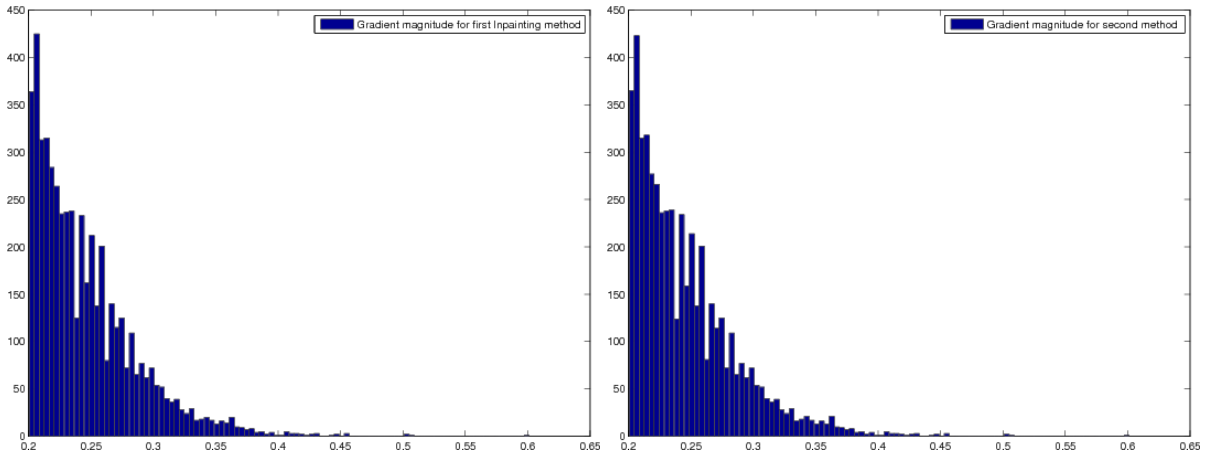


Figure 5: Comparisons of gradient magnitudes in inpainted regions. (a) when using energy term E_3^1, (b) when using energy term E_3^2. They are almost identical.

overlap in time. Here too the energy $E_2 + E_3^3 + E_4^2$ was used, with the same parameters as above. Figure 6 presents four frames of the degraded sequence, the inpainting results using energy (E_2, E_3^3, E_4^2), and a solver where the optical flow is computed with this energy while for the inpainting we use the lower order equation (equivalently setting the gradient parameter γ to 0 in E_3^3). Differences are non significative. This too substantiates the idea that the lower order inpainting equation is a good descent direction.

The third sequence, `Manon` is a real *color* sequence acquired by the first author with a mobile device, featuring the author's daughter. The original sequence was encoded with the 3G format, a simplified MPEG format for third generation mobile devices. It has been artificially degraded with red blotches and inpainted with the *vectorial form* of the energy (E_2, E_3^2, E_4^2) with $\lambda_2 = 0.1, \lambda_3 = 1, \gamma = 0.1$ and $\lambda_4 = 0.02$ a value 5 times smaller than in the other experiments, because a too large values caused problems in the recovery of hair motion among others. By its encoding and its nature, a face with non rigid motion, it presents some challenges. Three original frames, 7, 8 and 9, the corresponding degraded sequences as well as the inpainted ones are presented on Figure 7. The result is visually very good. Nevertheless, some serious problems were encountered at other locations of the sequence and Figure 8 presents some of the encountered problems at the 5th frame of the sequence, where a portion of the hair and of an eye are wrongly interpolated.

The fourth and last sequence, called `Frankenstein`, also taken from Kokaram's book, is a real degraded one, with 64 frames, for which no ground truth is known. In our experimentations we used only a subsequence of 21 frames, with frame 8 presenting, among others, a relatively large blotch on Frankenstein's hair. In figure 9 we show this frame, the detected blotch and its reconstruction. In figure 10 we show a close-up of the damaged hair of the character with the detected blotch and the reconstruction. Blotches were detected using the so called Rank Order Detector (ROD), as described in [33], modified for the optical flow algorithm. Fine texture details were very plausibly recreated.

7 Conclusion

In this paper we have introduced a generic variational formulation for joint recovery of motion and intensity in degraded image sequences, dealing both with noise and missing data. This generic formulation has been instantiated in several energy formulations, mostly based on known motion recovery approached. They give rise to system of partial differential equations, for motion and intensity. We have focused on the intensity ones and developed schemes to handle them numerically. We have validated our approach on a series of experiments. While they provide often excellent results, they are generally computationally demanding, especially due to higher order equations to solve for Inpainting. Is such complex equation necessary? We discussed that a simpler equation might still provide a good descent direction when minimizing a higher order energy, and we presented an experiment that substantiate this idea: there is a difference, but the result obtained with lower order equation is visually good. The possibility of using a lower order equation opens the door for much more efficient solvers: multigrid solvers where developed for optical flow by [40] as well as [13] and we are currently working on developing fast multigrid schemes for lower order inpainting equations. The general multiresolution optimization framework that we have used, decouples, at each pyramid level, the computation of the flows and the images. We are also investigating more coupled methods, also within the multigrid framework.

References

[1] L. Alvarez, R. Deriche, T. Papadopoulo, and J. Sanchez. Symmetrical dense optical flow estimation with occlusion detection. In A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, editors, *Proceedings of the 7th European Conference on Computer Vision*, pages 721–735, Copenhagen, Denmark, May 2002. Springer–Verlag.

[2] L. Alvarez, R. Deriche, J. Weickert, and J. Sànchez. Dense disparity map estimation respecting image discontinuities: A PDE and scale-space based approach. *Journal of Visual Communication and Image Representation, Special Issue on Partial Differential Equations in Image Processing, Computer Vision and Computer Graphics*, 13(1–2):3–21, 2002.

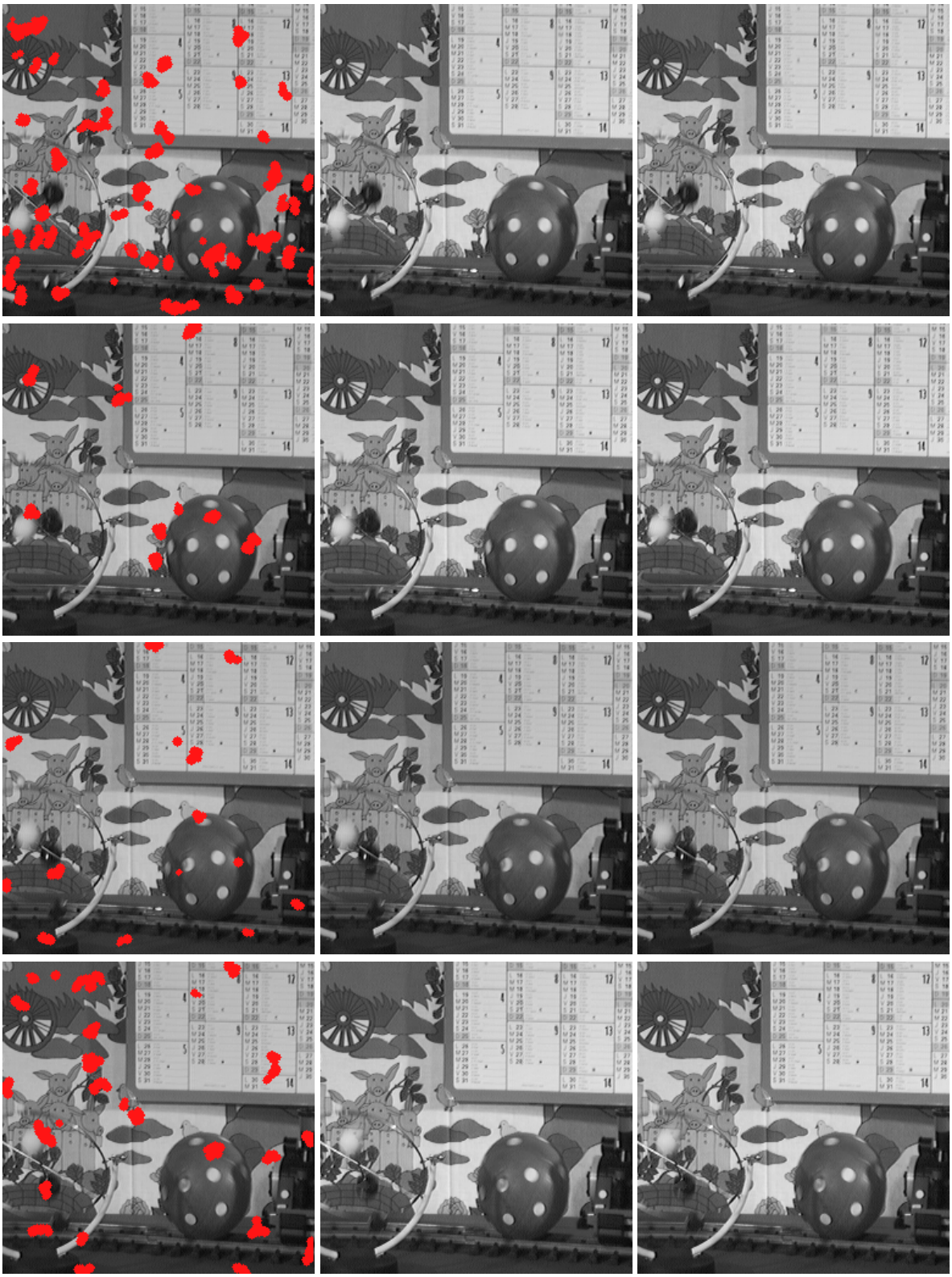


Figure 6: On the first column, frames 10, 11, 12 and 13 of the degraded `Mobile and Calendar` sequence. The second column shows the results of minimization of energy $E_2 + E_3^3 + E_4^2$ while the last column shows the results when the corresponding inpainting equation is replaced by the low order one.

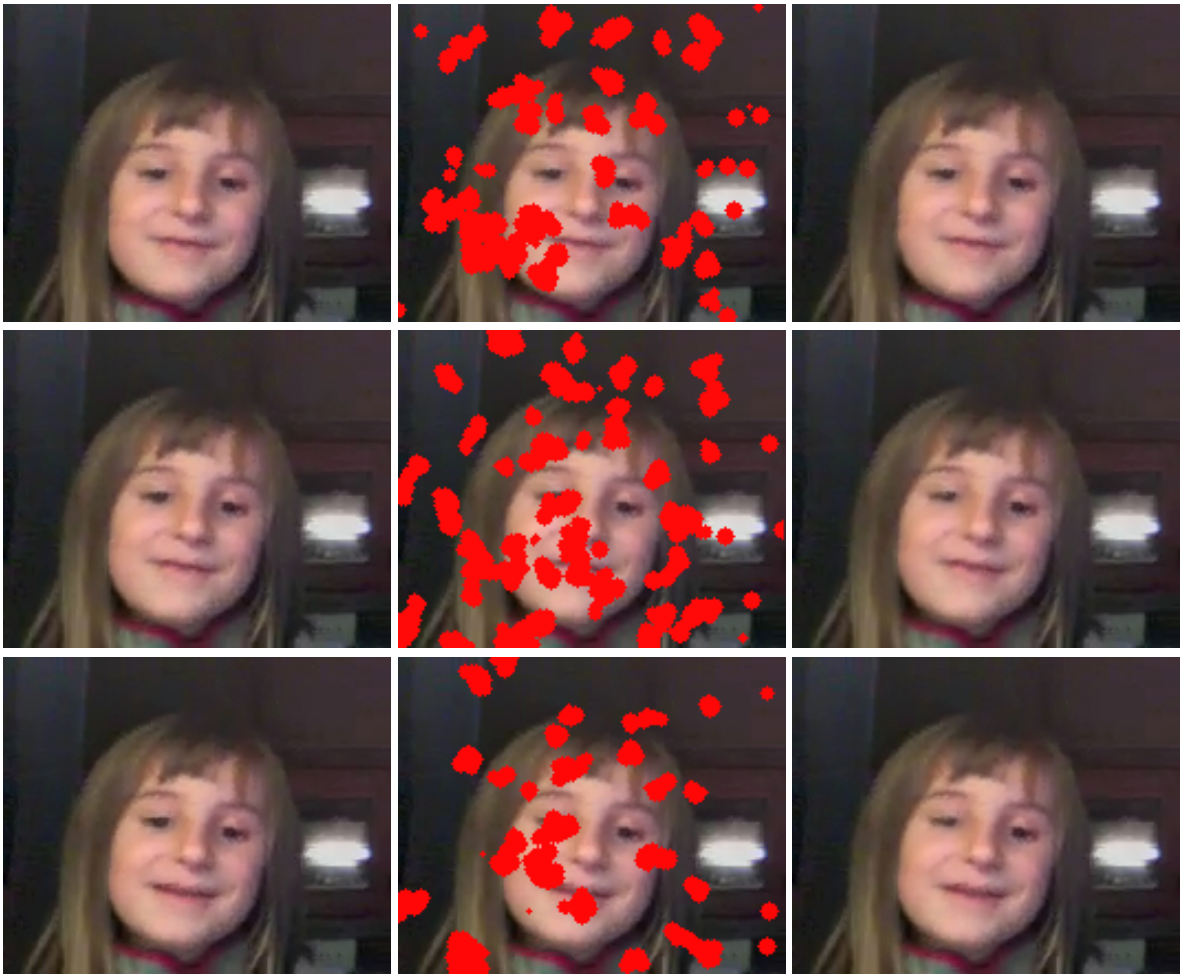


Figure 7: On the first column, frames 7,8 and 9 of the original `Manon` sequence. The second column shows the corresponding degraded frames and the last column the results of the color inpainting using Energy (E_2, E_3^2, E_4^2).

Figure 8: Original, degraded and inpainted frame 5 of the `Manon` sequence. Problems can be observed in the reconstruction of the left eye as well as part of the hair above it.

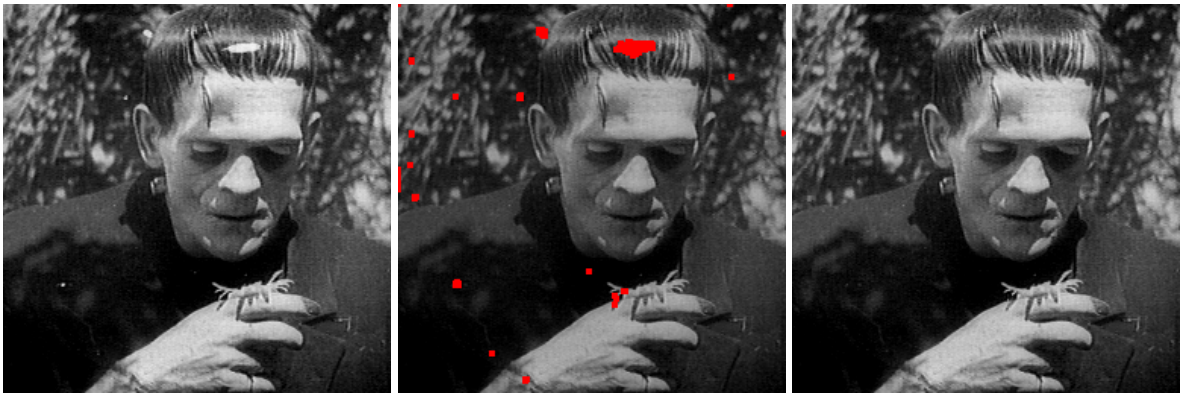


Figure 9: The `Frankenstein sequence`. From left to right: Frame 8, ROD detected defects and inpainting. by minimizing energy $E_2 + E_3^2 + E_4^2$

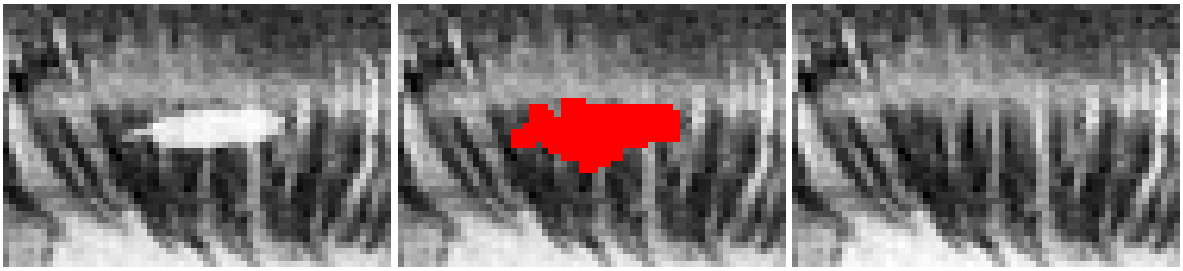


Figure 10: The `Frankenstein sequence`: Close-up of the hair blotch, its detection and its inpainting
.

[3] L. Alvarez, J. Weickert, and J. Sànchez. Reliable estimation of dense optical flow fields with large displacements. *The International Journal of Computer Vision*, 39(1):41–56, August 2000.

[4] G. Aubert, R. Deriche, and P. Kornprobst. Computing optical flow via variational techniques. *SIAM Journal of Applied Mathematics*, 60(1):156–182, 1999.

[5] G. Aubert and P. Kornprobst. A mathematical study of the relaxed optical flow problem in the space BV. *SIAM Journal on Mathematical Analysis*, 30(6):1282–1308, 1999.

[6] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations, Second edition*, volume 147 of *Applied Mathematical Sciences*. Springer, 2006.

[7] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing*, 10(8):1200–1211, 2001.

[8] D. Béréziat, I. Herlin, and L. Younes. A generalized optical flow constraint and its physical interpretation. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 487–492, Hilton Head Island, South Carolina, June 2000. IEEE Computer Society.

[9] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In Kurt Akeley, editor, *Proceedings of the SIGGRAPH*, pages 417–424. ACM Press, ACM SIGGRAPH, Addison Wesley Longman, 2000.

[10] J. Boulanger, C. Kervrann, and P. Bouthemy. Space-Time Adaptation for Patch-Based Image Sequence Restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1096–1102, 2007.

[11] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In T. Pajdla and J. Matas, editors, *Proceedings of the 8th European Conference on Computer Vision*, volume 4, pages 25–36, Prague, Czech Republic, 2004. Springer–Verlag.

[12] A. Bruhn and J. Weickert. Towards Ultimate Motion Estimation: Combining Highest Accuracy Optical Flow with Real-Time Performance. In *International Conference on Computer Vision*, volume 1, pages 749–755, Beijing, China, 2005. IEEE Computer Society Press.

[13] A. Bruhn, J. Weickert, T. Kolhberger, and C. Schnörr. A Multigrid Platform for Real-Time Motion Computation With Discontinuity-Preserving Variational Methods. *The International Journal of Computer Vision*, 70(3):257–277, 2006.

[14] A. Buades, B. Coll, and J. M. Morel. Nonlocal Image and Movie Denosing. *The International Journal of Computer Vision*, 76(2):123–139, 2008.

[15] A. Buades, J. Delon, Y. Gousseau, and S. Masnou. Adaptive blotch detection for film restoration. In *Proceedings of International Conference on Image Processing*, pages 3317–3320, Hong-Kong, China, September 2010. IEEE, IEEE.

[16] T. Chan and J. Shen. Mathematical models for local nontexture inpainting. *SIAM journal of appl. Math*, 62(3):1019–1043, 2002.

[17] T.F. Chan and J. Shen. Non-texture inpainting by curvature-driven diffusions (cdd). *J. Visual Comm. Image Rep.*, 12(4):436–449, 2001.

[18] J.P. Cocquerez, L. Chanas, and J. Blanc-Talon. Simultaneous inpainting and motion estimation of highly degraded video-sequences. In *Scandinavian Conference on Image Analysis*, pages 523–530. Springer-Verlag, 2003. LNCS, 2749.

[19] A. Criminisi, P. Pérez, and K. Toyama. Region filling and object removal by exemplar-based inpainting. *IEEE Trans. Image Processing*, 13(9):1200–1212, 2004.

[20] L. d'Amore, L. Marcellino, and A. Murli. Image Sequence Inpainting: Towards Numerical Software for Detection and Removal of Local Missing Data with Motion Estimation. *Journal of Computational and Applied Mathematics*, 198(2):396–413, January 2007.

[21] Alexei A. Efros and William T. Freeman. Image quilting for texture synthesis and transfer. *Proceedings of SIGGRAPH 2001*, pages 341–346, August 2001.

[22] Alexei A. Efros and Thomas K. Leung. Texture synthesis by non-parametric sampling. In *IEEE International Conference on Computer Vision*, pages 1033–1038, Corfu, Greece, September 1999.

[23] L.C. Evans. *Partial Differential Equations*, volume 19 of *Graduate Studies in Mathematics*. Proceedings of the American Mathematical Society, 1998.

[24] L. Florack, W. Niessen, and M. Nielsen. The intrinsic structure of optic flow incorporating measurement duality. *The International Journal of Computer Vision*, 27(3):263–286, 1998.

[25] S. Gallot, D. Hulin, and J. Lafontaine. *Riemannian Geometry*. Springer-Verlag, 1990.

[26] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[27] G. Gilboa, Y. Y. Zeevi, and N. Sochen. Image Sharpening by Flows Based on Triple Well Potential. *Journal of Mathematical Imaging and Vision*, 20(1–2):121–131, January–March 2004.

[28] H. Grossauer. Inpainting of Movies Using Optical Flow. In O. Scherzer, editor, *Mathematical Methods in Registration for Applications in Industry and Medicine*, pages 151–164. Springer, 2006.

[29] M. Irani and S. Peleg. Motion analysis for image enhancement: resolution, occlusion, and transparency. *Journal on Visual Communications and Image Representation*, 4(4):324–335, 1993.

[30] S. H. Keller, F. Lauze, and M. Nielsen. Motion Compensated Video Super Resolution. In F. Sgallari, A. Murli, and N. Paragios, editors, *Proceedings of the First International Conference on Scale Space and Variational Methods in Computer Vision*, volume 4485 of *LNCS*, pages 801–812, Berlin, 2007. Springer.

[31] S. H. Keller, F. Lauze, and M. Nielsen. Deinterlacing Using Variational Methods. *IEEE Transactions on Image Processing*, 17(11):2015–2028, 2008.

[32] S. .H. Keller, F. Lauze, and M. Nielsen. Temporal Super Resolution. In M. Mrak, M. Kunt, and M. Grgic, editors, *High-quality Visual Experience: creation, processing and interactivity of high-resolution and high-dimensional video signals (to appear)*, Signal and Communication Technologies. Springer, 2010.

[33] A. Kokaram. *Motion Picture Restoration: Digital Algorithms for Artefact Suppression in Degraded Motion Picture Film and Video*. Springer, 1998.

[34] A. Kokaram. On Missing Data Treatment for Degraded Video and Film Archives: A Survey and a New Bayesian Approach. *IEEE Transactions on Image Processing*, 13(3):397–415, March 2004.

[35] A. Kokaram, R. Morris, W. Fitzgerald, and P. Rayner. Interpolation of Missing Data in Image Sequences. *IEEE Transactions on Image Processing*, 4(11):1509–1519, 1995.

[36] F. Lauze and M. Nielsen. A variational algorithm for motion compensated inpainting. In S. Barman A. Hoppe and T. Ellis, editors, *British Machine Vision Conference*, volume 2, pages 777–787. BMVA, 2004.

[37] M. Lysaker, A. Lundervold, and X-C. Tai. Noise removal using fourth-order partial differential equations with applications to medical magnetic resonnance images in space and time. *IEEE Transactions on Image Processing*, 12(12):1579–1590, 2003.

[38] S. Masnou and J.M. Morel. Level lines based disocclusion. *International Conference on Image Processing*, III:259–263, 1998.

[39] E. Mémin. *Estimation du flot optique: contributions et panorama des différentes approches*. IRISA+IFSIC, July 2003. Document d'habilitation à diriger des recherches.

[40] E. Mémin and P. Pérez. Hierarchical estimation and segmentation of dense motion fields. *The International Journal of Computer Vision*, 46(2):129–155, 2002.

[41] D. Mumford. Bayesian rationale for the variational formulation. In *Geometry-Driven Diffusion In Computer Vision*, pages 135–146. Kluwer Academic Publishers, 1994. Computational Imaging And Vision.

[42] H.H. Nagel and W. Enkelmann. An investigation of smoothness constraint for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:565–593, 1986.

[43] M. Nikolova. Weakly Constrained Minimization: Application to the Estimation of Images and Signals Involving Constant Regions. *Journal of Mathematical Imaging and Vision*, 21(2):155–175, 2004.

[44] K. A. Patwardhan, G. Sapiro, and M. Bertalmio. Video Inpainting of Occluding and 0ccluded 0bjects. In *International Conference on Image Processing*, volume II(2), pages 69–72, 2005.

[45] K. A. Patwardhan, G. Sapiro, and M. Bertalmio. Video Inpainting Under Constrained Camera Motion. *IEEE Transactions on Image Processing*, 16(2):545–553, Feb 2007.

[46] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[47] A. Verri and T. Poggio. Motion field and optical flow: qualitative properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(5):490–498, 1989.

[48] X. Wang and M. Mirmehdi. Hmm based archive film defect detection with spatial and temporal constraints. In *Proceedings of BMVC*. BMVA, 2009.

[49] X. Wang and M. Mirmehdi. Archive film restoration based on spatiotemporal random walks. In *Proceedings of the 11th European Conference on Computer Vision*, volume 6315 of *LNCS*, pages 478–491, Heraklion, Greece, September 2010. Springer.

[50] J. Weickert and C. Schnörr. A Theoretical Framework for Convex Regularizers in PDE-Based Computation of Image Motion. *The International Journal of Computer Vision*, 45(3):245–264, December 2001.

[51] Y. Wexler, E. Shechtman, and M. Irani. Space-Time Completion of Videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):463–467, 2007.

[52] R.P. Wildes, M.J. Amabile, A.M. Lanzillotto, and T.S. Leu. Physically based fluid flow recovery from image sequences. In *International Conference on Computer Vision and Pattern Recognition*, pages 969–975, 1997.