1 Action points

1.1 Questions concerning application project A1:

1.1.1 Background and clinical relevance

How do images of grey and white matter (volumes, cortical thickness) look like in MRI and PET and in CT? (Including typical volume size, number of projections, voxels, detector resolution, distances (source-sample, sample-detector), grey values, etc)

On what image features is AD diagnosed?

E. g. regional measure of atrophy: hippocampus, amygdala; global measure: total amount of cerebrospinal fluid (CSF), grey and white matter. Estimated via brain segmentation using popular, magnetic resonance imaging (MRI)-based Software as FreeSurfer, FSV, or CIVET (Corticometry Analysis tool, CIVET: seems to refer to brain-imaging pipeline within brainstorm software package?)

What's the relative difference in amount of CSF, grey matter (GM), or white matter (WM) one is looking for?

Atrophy of what? Grey or white brain matter, or brain matter in general?

What is the physical origin (difference) related to the image features in MRI, PET, and CT? How does contrast arise in grey and white matter?

What are the neuroanatomical phantom? How do they look like? Do they include hippocampus, amygdala, etc.

Parameters for helical acquisition geometry? longitudinal translation range, number of projections, full angle of rotation, number of 2D pixels, size of detector, distance source sample, distance sample detector,

1.1.2 Project description

Prior model

Complexity in low complexity refers to what? What is the mathematical measure of complexity, what quantity evaluates complexity?

Either by measures for sparsity or for entropy. Image features are associated with a suitable notion of image complexity which is encoded by a specific choice of $\mathcal{S}$.

How relate grey-white-matter contrast to low complexity models for edges? Sparsity. Entropy?

Forward model

Standard forward model of CT employs simple Beer-Lambert's law for the interaction of X. What is the origin of the attenuation? Photoabsorption, strong elastic scattering, (inelastic) Compton scattering, ..

What would a more refined model include?

How is the attenuation coefficient treated? As an effective coefficient over an energy band width? Value at peak energy?

What is are the preprocessing steps for the data acquired by a CT scanner?

What is the data format? Is it free or proprietary? Are readers available for Matlab or python?

Reconstruction scheme

Sparsity of gradient as low-complexity representation -> Try TV-regularisation in two variants Bregman-TV and lambda-TV.

Wavelet/Shearlet compression: test sparse l1-regularisation.

Implement entropy type of complexity measures using an edge-prior given by lambda-tomography.

How is the entropy-type of complexity measures defined? Where is it useful? What are the drawbacks?

1.2 To do

Consider low-complexity models for edges. Based on sparsity of the gradient and shearlets, of compression by learned dictionaries. The low-complexity representations need to be evaluated against a set of 3D images of neuroanatomical phantoms relevant for 3D?

Develop fast implementation of backprojection for certain scanning geometries? Relevant geometries for CT is helical image acquisition? Cone beam reconstruction works using All Scale Tomographic Reconstruction Antwerp (ASTRA) for data simulated in ASTRA. GATE shows artefacts. Origin of these is unclear.

Variants of TV-regularisation: Bregman-TV and lambda-TV.

Reconstruct data from GATE.

Validation of reconstruction via brain segmentation using MRI-based algorithms (FreeSurfer, FSL, CIVET)).

Installing FreeSurfer: Installation worked, Installation testing successful.

Installing FSL: For Debian/Ubuntu users a repository is provided by Neurodebian. Installation from this site doesn't work because it wants to remove CUDA packages. Try installation by compilation from source code.

Installing git: Done. Create account: Done.

Acquire example script from Kati on how to use the SPOT operator.

Check ASTRA toolbox on using Parker weighting for cone-beam reconstructions w.r.t. if this is the reason for false reconstructions.

1.3 Deliverables

1.3.1 Characterise application

Action 35: Document the specific characteristics of the inverse problem associated with application A1.

Deliverable: Report outlining characteristics of the inverse problem in application A1.
Deadline: 2015-03-10

What image features are of clinical relevance? What is the computational complexity associated with the reconstruction (influencing the regularisation to choose)?

Protocol for head CT typically involves FFS, i. e. a periodic movement of the source in plane (xy-FFS), along the longitudinal direction (z-FFS), or both (xyz-FFS). Thereby, instead of one data set twice, thrice, or four times the number of projections of a a non-FFS scan are acquired .This is improves sampling allowing for a higher spatial resolution at the same pitch factor or keeping the resolution constant at a higher pitch factor. z-FFS also reduces helical acquisition artifacts. The article introducing z-FFS uses rebinning of the projection and subsequent parallel-beam reconstruction. ASTRA's projector for cone-beam geometry allows for arbitrary positioning of the detector reference point and the source and arbitrary orientations of the detector. Knowledge of source positions for FFS acquisition, reconstructions using ASTRA should not require a projection weighting. To KI medical physicist Daniel Thor these positions are not available. The xy-FFS is automatically triggered by the acquisition protocol, usually when pitch factor are ≤ 1 which is the case for head. The z-FFS can be triggered manually which is also common for head CT according to the acquisition protocols obtained from him. At this stage the FFS positions are not accessible from Daniel Thor, since data processing and reconstruction is completely hidden by the CT scanner. However, real xyz-FFS data from an abdomen phantom is provided by Mayo clinic including FFS position values.

1.3.2 Forward model software

Needs: Compute the forward model and the adjoint of its derivative.

Make use of existing software suites such as ASTRA and/or NiftyRec (or TomoPy) for application A1. Combing through ASTRA lib. Need to contact ASTRA developer, but first need specify issues. Check python interface of ASTRA

Action 37: Download, install, and test ASTRA and NiftyRec.

Documentation estimating development efforts needed for using these software suites as forward model application. Assess which forward models and data acquisition geometries are supported.

Considered software suites: ASTRA, NiftyRec (NiftyRec), and Tomographic Reconstruction in Python (TomoPy)

TomoPy only supports parallel-beam geometries and is thus not considered. Due to problems running **nifty!** (**nifty!**), it was not considered so far. However, the latest release of **nifty!** seems to work and show a good match of the backprojector being the adjoint of the (derivative of the) forward operator, i. e. $\frac{<Ax,y>}{x,A^*y} \approx 1 \times 10^{-5}$. If this holds one should reconsidered **nifty!** for CT applications.

The RTK package based on ITK also provides forward and backward projector for cone beam geometries. Installation of RTK failed so far.

Geometries supported in general by ASTRA:

- parallel 2D
- parallel 3D
- fanbeam
- circular conebeam
- helical conebeam

Availability of geometries depends on the processing unit.

Geometries for central processing unit (CPU):

- Parallel beam (2D) with weights
  - line: The weight of a ray/pixel pair is given by the length of the intersection of the pixel and the ray, considered as a zero-thickness line.
  - strip: The weight of a ray/pixel pair is given by the area of the intersection of the pixel and the ray, considered as a strip with the same width as a detector pixel.
  - linear: A ray is traced through successive columns or rows (depending on which are most orthogonal to the ray). The contribution of this column/row to this ray is then given by linearly interpolating between the two nearest volume pixels of the intersection of the ray and the column/row.

    This is also known as the Joseph kernel, or a slice-interpolated kernel.
- Fan beam (2D) with weights
  - line: The weight of a ray/pixel pair is given by the length of the intersection of the pixel and the ray, considered as a zero-thickness line.
  - strip: The weight of a ray/pixel pair is given by the area of the intersection of the pixel and the ray. The ray is considered as a 2D cone from the source to the full width of the detector pixel. The projector can only be used with the fanflat geometry.

    Remark: This mathematical model does not properly take into account the fan beam magnification effect.

Geometries for graphics processing unit (GPU):

- parallel beam (2D)
- fan beam (2D)

- parallel beam (3D)
- cone beam (3D)

The forward and backward projection for the above geometries is successfully tested. However, the current version exhibits several inconsistencies

1.3.3 Dictionaries

Low complexity representations of 3D images.
Consider software framework SPArse Modeling Software (SPAMS):

Installation

Installation successful. All test scripts are running. In the case of a segmentation fault error running SPAMS test script, remove line from `start_spams.m` or MATLAB's `startup.m`:
`setenv('MKL_DYNAMIC','NO')`

1.3.4 Visualisation software

Visualisation software suitable for general purposes: arrayShow tool (Matlab), ImageVis3D, MeVisLab, medInria, GIMIAS, Fiji, icy, VTK base volume renderers like ParaView.
In particular for application A1 more specialised visualisation software is required to analyse neuroanatomical structures: FreeSurfer, FSL, CIVET.
Action 41: Determine suitable visualisation software for assessing reconstruction quality relevant for application A1.

Deadline: 2015-03-15
Visualisation software relevant for application A1 is FreeSurfer.
To do: Acquire GATE data of advanced brain phantom and test visualisation on reconstructed volumes.

1.3.5 Reconstruction in a highly simplistic setting

Test software of Section 1.3.2 by performing a reconstruction from simplistic simulated data using a method already available within the forward-model software of Section 1.3.1.
Action 45:

Deadline: 2015-04-01
At the moment, NiftyRec is disregarded as a forward-model software for the application A1, i. e. transmission tomography due to its strange behaviour. Therefore, focus is on ASTRA. Currently checking how to implement in operator disrectisation library (ODL) framework, more specifically considering scaling.

Simplistic data from GATE (simple spheres) is currently provided as Matlab files. Thus data reading is trivial and reconstructions should work in principal. Problem so far is the occurrence of artefacts of unknown origin.

To do: What is simplistic data is still realistic enough for the envisage application goal? Data size: number of detector pixels (2D) and number of projections. Size of neuroanatomical structures.

1.3.6 Compute forward model and the adjoint of its derivative

Action 51: Develop software components within the platform for variational regularisation (action 50) for computing the forward model for application A1 making use of components from forward model software in action 37. The data acquisition geometries need to support those that arise in simplistic simulated data from action 44.

Deliverables: Software components within the platform for variational regularisation (action 50) for computing the forward model for application A1 with proper coupling to relevant routines provide by forward model software in action 37. Data acquisition geometries should match those for simplistic simulated data from action 44.

Deadline: 2015-06-01

Action 54: Derive analytic expressions for the adjoint of the derivative of the forward model in application A1. Develop software components for within the platform for variational regularisation (action 50) for computing this derivative consistent with routines for the forward model in action 51, typically making use of components from forward model software in action 37. The data acquisition geometries supported need to correspond to those that arise in simplistic simulated data from action 44.

Deliverables: Software components within the platform for variational regularisation (action 50) for computing the adjoint of the forward model for application A1 with proper coupling to relevant routines provide by forward model software in actions 37 and 51. Data acquisition geometries should match those for simplistic simulated data from action 44.

Deadline: 2015-06-15

Action 58: Develop software components within the framework in action 57 to compute (2.1) and its gradient in the context of application A1. Use software components from actions 51, and 54 for the delegated computation. Ensure resulting routines are computationally feasible for clinically relevant problem sizes and time constraints (action 34).

Deliverables: Software components within the framework in action 57 for computing (2.1) and its gradient in the context of application A1.

Deadline: 2015-08-15

Action 67: Consider energy functionals of ‘p-type w.r.t. a dictionary as in (A.9). Derive analytic expressions for its gradient. Using the framework from action 61, implement routines for evaluating (A.9) and its gradient.

Deliverables: Expression for the gradient of (A.9). Software components within the framework in action 61 for computing (A.9) and its gradient. Here you may assume there are software components for sparse representation and synthesis.

Deadline: 2015-09-01

Action 68: Implement classical Tikhonov regularisation for application A1. The implementation should constitute a component within the platform for variational regularisation (action 50) and it should be applicable to reconstruct from simplistic simulated data generated for application A1 (action 44). Furthermore, it should make use of software components from actions 58 and 62. Make sure to use an optimisation scheme that is computationally feasible bearing in mind the problem size and time constraints associated with application A1 (action 34).

Deliverables: Implementation of classical Tikhonov regularisation for application A1 within the platform for variational regularisation (action 50) that is computationally feasible for clinically relevant problem sizes and time constraints (action 34).

Deadline: 2015-10-01

Action 71: Implement TV regularisation for application A1. The implementation should constitute a component within the platform for variational regularisation (action 50) and it should be applicable to reconstruct from simplistic simulated data generated for application A1 (action 44). Furthermore, it should make use of software components from actions 58 and 63. Make sure to use an optimisation scheme that is computationally feasible bearing in mind the problem size and time constraints associated with application A1 (action 34).

Deliverables: Implementation of TV regularisation for application A1 within the platform for variational regularisation (action 50) that is computationally feasible for clinically relevant problem sizes and time constraints (action 34). Deadline: 2015-11-01

2 Software, hardware, data

2.1 Dictionaries

Low complexity representations of 3D images requires software for representing images in terms of suitable dictionaries. Consider software framework SPAMS.

2.1.1 SPAMS

Installation successful. All test scripts are running. In the case of a segmentation fault error running SPAMS test script, remove line from `start_spams.m` or MATLAB's `startup.m`:

```
setenv('MKL_DYNAMIC','NO')
```

2.2 GPUs

Working station desktop has two GPUs. One for graphical output and one for computations. Index of GPU devices is found using `nvidia-smi`:

GPU	Name	Bus-ID	Memory
0	GeForce GTX 980	0000:02:00.0	4095MiB
1	Quadro K620	0000:03:00.0	2047MiB

Tabelle 2.1: graphics processing unit.

The same indices are used by ASTRA and python interface to the ASTRA toolbox (PyASTRA). Pay attention that the order of indices is reversed in NVIDIA (NVIDIA)'s graphical X-server-settings interface `nvidia-settings`. Index of working-horse GPU in ASTRA and PyASTRA is 0.

2.3 ASTRA

2.3.1 Matlab

For ASTRA to compile mex files compiler version 4.7 is needed. As a quick and dirty hack change the symbolic links of `gcc` and `g++`:

```
sudo rm /usr/bin/gcc; sudo ln -s /usr/bin/gcc-4.7 /usr/bin/gcc
sudo rm /usr/bin/g++; sudo ln -s /usr/bin/g++-4.7 /usr/bin/g++
```

2.3.2 PyASTRA

E. g. install toolbox with following options:

```
sudo ./install.sh -i  /Software/ASTRA/astra-1.5/include/ -l /usr/local/astra/
-p /usr/bin/python -c /usr/local/cuda
```

ASTRA library needs to be included in `LD_LIBRARY_PATH`. E. g. add following line to '.profile' (or '.bashrc'):

```
export LD_LIBRARY_PATH=/usr/local/astra/lib:LD_LIBRARY_PATH
```

2D volume geometries provide optional argument defining the extend of the volume. 3D geometries assume an isotropic unit voxel size and express all length scales in terms of the voxel size. For 2D geometries: if the optional argument is given do NOT rescale length scales.

2.3.3 Test of geometries

ASTRA: Parallel-, fan-, and cone-beam reconstruction works for data simulated by ASTRA. Phantom data provided by GATE (GATE) exhibits artefacts when using fan- or cone-beam reconstruction which originate from a false direction of rotation. Multiplication of rotation angles by -1 solves the problem. Fan-beam reconstruction of the walnut data works perfectly. Helical CT data from GATE not yet available and reconstruction not tested yet. Helical reconstruction on phantom data generated by ASTRA successfully tested. Test of helical reconstruction using GATE data pending.

2.4 NiftyRec

NiftyRec: Modification of source code (`_tt_line_backproject_ray_gpu_kernels.cu`) before compilation is needed in order to use the backprojection function with GPU. Otherwise the function for the backprojection using GPU is not working and Matlab crashes. However, NiftyRec shows a strange behaviour: Using CPUs test reconstruction are working. Using GPU, test reconstruction (`tt_demo_mlem_ conebeam`) only gives flat reconstructions, moreover reconstructions are slower than using CPUs. GPU option for forward projection (`tt_project_ray_mex`) seems to be broken, computation is always executed on GPU.

2.5 TompPy

Developed at the **aps!** (**aps!**) in Chicago for synchrotron beamline the software package seems to support only parallel-beam geometry and is thus excluded from further considerations.

2.6 SPAMS

SPAMS is an optimisation toolbox for solving various sparse estimation problems.

- Dictionary learning and matrix factorization (NMF, sparse PCA, ...)
- Solving sparse decomposition problems with LARS, coordinate descent, OMP, SOMP, proximal methods
- Solving structured sparse decomposition problems (l1/l2, l1/linf, sparse group lasso, tree-structured regularisation, structured sparsity with overlapping groups, ...)

2.7 DTIcode

DTI abbreviates diffusion tensor imaging.

Written 2012-2014 by Tuomo Valkonen <tuomov@iki.fi>, University of Graz & University of Cambridge.

This repository includes code for TGV^2 and TV denoising and reconstruction from sparse k-space data. For problems with linear operators, the Chambolle-Pock [3] modified primal dual hybrid gradientn method (PDHGM) is used. For problems with non-linear, the non-linear PHDHGM of [5] and the Gauss-Newton method are supported. Optional Bregman iterations may be applied to many problems. The code is parallelised using OpenMP. There is also OpenACC GPU support, but at the moment it is not actively maintained.

For linear operators the system is stored, see userguide.pdf in subfolders doc. Since the linear operator has to be stored and there is no option to provide the adjoint operator (transpose system matrix), it is unlikely that the algorithmic routines of this software package can be adapted for the operator disrectisation library (ODL) framework without major modification of the source code.

Status: This software is no longer considered.

2.7.1 Installation

Misleading description of installation procedure. Working procedure:
First install `libtu` and then `dticode`:
Install `libtu`:

1. Set `HAS_SYSTEM_ASPRINTF=0` in `system.mk`
2. Optionally: Set prefix.
3. Use `makedepend` instead of `make depend`
4. Optionally: `sudo make checkinstall`

Install `dticode`:

1. Copy `locale.mk.in` to `locale.mk`
2. In `locale.mk` set CUDA library

3. In `locale.mk` set libtu:
   `LIBS += -L$(HOME)/Software/DTIcode/dticode/libtu/lib`
   `DEFINES += -I$(HOME)/Software/DTIcode/dticode/libtu/include`
4. In `locale.mk` set mex locaton for Matlab.
5. `make`
6. optionally: `sudo checkinstall`

2.8 Visualization software

2.8.1 CIVET

Refers to a human-brain image processing pipeline for fully-automated corticometric, morphometric and volumetric analyses of magnetic resonance (MR) images.

2.8.2 FreeSurfer

Installation successful. Compilation from source code needed. Debian/Ubuntu repositories not working. Installation tests successful (freeview (volume & surface viewer), tkmedit (volume viewer), tksurfer (surface viewer), qdec, recon-all -all).

2.8.3 FSL

Compilation form source code needed. After building of projects following error message occurs:

```
!!ERROR in BUILD!!
  Could not make the following projects successfully:
    miscvis
```

Nevertheless software is running, but if missing function miscvis results in software failure later on is unclear.

2.9 To Sort

ASTRA and pyastra offer to explicitly retrieve the weight matrix encoded by its projector. This is only implemented for 2D CPU projectors.

However, this does not work for CUDA projectors which do not return a weight matrix, MATLAB and not available in PyASTRA. 2D projectors for CPU are available in MATLAB.

3 Data

3.1 Phantoms

3.1.1 Kati box

3D grey scale phantom
shape = (100, 100, 100)
Remark by David: voxelized phantom similar to what we discussed. The `Kati_box` just shows a label map (0,15-85 and 200) for which materials are attached. Of more use is perhaps `Kati_box_HU` which are the corresponding HU for the phantom.

Sinogram

Use same projection data as Kati to have same identical noise component.

Sinogram simulated from the 3D gray scale phantom. If you want to avoid inverse crime reconstruct with voxel size less than 100 (the original phantom was 100 x 100 x 100 hence the data is simulated with N=100) The standard deviation of the additive white Gaussian noise is 1%. And simulated with 'parallel3d'.

3.2 Data for Alzheimer's disease

Currently, the clinical standard of diagnosis of AD is the examination by the physician. MRI images are only supportive. Hereby, only the visual inspection of images at different ages are compared. Numerical values of a decrease in GM or cortical thickness are not FDA approved bio markers and so far only used in clinical research. Besides using the visual inspection of reconstructions from CT scans instead of MRI, the clinicians would like use such bio marker in order to support the diagnosis of AD.

3.2.1 Open questions?

Determine normal dose levels, i.e. usually employed dose at a standard CT-scan, and high-dose level, i.e. where grey and white matter can be differentiated.

Estimate counts/pixel that are sufficient to determine the difference of the attenuation coefficient between grey and white matter using a simple phantom and geometry (e. g. two wires of grey and white matter in a sphere-like water environment. Consider dependence on acquisition and reconstruction protocol (cone <-> helical, pixel size, number of pixels, projections, focal spot size, flying focal spot, preprocessing, ...). Non Poisson noise? Which? From dark current. Skull-structure surrounding the brain influences the dose needed.

Phantoms: Decide on phantom including the relevant features of which, together with Mamo and David to make sure it is feasible to use them for GATE simulations and with the physicians to make sure it is relevant for AD. GATE phantoms for normal dose CT and high dose CT. For larger phantoms use Jonas' simulation software. Documentation on phantoms. Clarify if ELEKTA MC code can be used.

Define set-up. To start with use the simplest but still reasonable setup. Ideally one should use a cone-beam geometry with helical acquisition and curve-detectors, and typical parameters for detector dimensions, number of projections, etc.

Reconstruct available (and relevant) data using available methods. FDK, SIRT, Landweber, positivity-constrained Chambolle-Pock with Tikhonov or TV.

PCXMC - A Monte Carlo program for calculating patient doses in medical x-ray examinations

Publisher's description PCXMC is a computer program for calculating patients' organ doses and effective doses in medical x-ray examinations (radiography and fluoroscopy). The doses are calculated in 29 organs and tissues and the program calculates the effective dose with both the present tissue weighting factors of ICRP Publication 103 (2007) and the old tissue weighting factors of ICRP Publication 60 (1991). The program incorporates adjustable-size paediatric and adult patient models and allows a free choice of the x-ray examination technique.

3.2.2 Current status

Check Siemens webpage for X-ray spectrum simulator.

Start with circular cone-beam acquisition, then refine to helical acquisition curve.

What is gain image? Ask Jonas.

What is the dynamic range of a real CT-scan? Ask Carlos or Daniel Thor. Parameters of the used protocols are not available. David will get information about that from Carlos.

4 Optimisation

4.1 General notes and things to do

iteration as a Check for convergence/semi-convergence properties, optimality conditions,

4.2 Duality

[BL15] Distinguish hard from easy problems not by non-linearity but by non-convexity, at least in linear optimization, see interior point revolution (barrier methods,...). For convex sets topological, algebraic, and geometric notions often coincide. The dual perspective yields new insights (statistically or information theoretically) in image processing problems and faster algorithms.

An instance of algorithms exploiting duality is denoted proximal algorithms. These are a standard tools for solving non-smooth, constrained, large-scale or distributed optimisation problems as Newton's method is for solving unconstrained smooth problems of modest size. The base operation is the evaluation of the proximal operator of a function. This itself involves solving a small optimisation problem which often admit closed-form solutions or are solvable quickly with standard or simple specialised methods [PB14].

Using primal-dual algorithms involving proximal methods thus allows to minimise certain object functionals, such as TV, without prior smoothing.

Consider typical optimisation model in imaging:

$$\begin{aligned} &\underset{x\in C\subset X}{\text{minimize}} && I_\varphi(x) \\ &\text{subject to} && Ax \in D \,. \end{aligned} \tag{4.1}$$

X and Y are real Banach space with continuous duals X^* and Y^*, C and D are closed convex sets, $A : X \to Y$ is a continuous linear operator, and the integral functional $I_\varphi(x) := \int_T \phi(x(t))\mu(dt)$ is defined on some vector subspace $L_p(T, \mu)$ of X with μ being a complete totally finite measure on some measure space. The integral operator is an entropy with integrand $\varphi : \mathbb{R} \to]-\infty, \infty[$ a closed convex function.

The algorithmic approach to applications implicit in the variational formulation of 4.1 has a correspondence to the feasibility approach to problems which considers the problem to find a point x lying in the intersection of constrained sets:

$$\text{find } x \in C \cap S \quad \text{where} \quad S := \{x \in X \mid Ax \in D\} \,. \tag{4.2}$$

When the intersection is quite large one wishes to find a point in the intersection which is in some sense closest to a reference point x_0. To find the best approximation is done by the objective in 4.1, namely to pick the element of $C \cap S$ with the desired properties.

Fenchel conjugate

Also called Legendre-Fenchel conjugate. The Fenchel conjugate is to convex analysis what the Fourier transform is to harmonic analysis. X is a normed space with dual $X*$ unless stated otherwise. The Fenchel conjugate $f^* : X^* \to [-\infty, \infty]$ of a mapping $f : X \to [-\infty, \infty]$ is defined as

$$f^*(x^*) = \sup_{x \in X} \{\langle x^*,\ x\rangle - f(x)\} \tag{4.3}$$

As a supremum of affine functions the conjugate is always convex.

4.2.1 Fenchel duality

Let X and Y be Banach spaces, $f : X \to [-\infty, \infty]$ and $g : X \to [-\infty, \infty]$, and $A : X \to Y$ a bounded linear mapping. The Fenchel problems define the primal and dual variables p, $d \in [-\infty, \infty]$:

$$\begin{aligned} p &= \inf_{x \in X} f(x) + g(Ax) \\ d &= \sup_{y^* \in Y^*} \ -f^*(A^* y^*) - g^*(-y^*)\ . \end{aligned} \tag{4.4}$$

These values satisfy the weak duality $p \geq d$. If f, g are convex and satisfy either

$$0 \in \text{core}(\text{dom } g - A \text{ dom } f) \quad \text{with } f \text{ and } g \text{ lower-semi-continuous (lsc)}\ , \tag{4.5}$$

or

$$A \text{ dom } f \cap \text{cont } g \neq \emptyset\ , \tag{4.6}$$

the $p = d$, and the supremum to the dual problem is attained.

The Fenchel duality allows to reverse minimisation and maximisation after rewriting (parts of) the cost functional in dual variables using convex conjugates:

$$\min_x \sup_u S(x, u) = \sup_u \min_x S(x, u)\ , \tag{4.7}$$

where u is the dual variable/value to the primal x defined via the Fenchel problems. S is a very simple function of x.

Calculus of Fenchel conjugation required to compute the conjugate of sums of functions.

A proximal method requires besides computing the convex conjugate and the proximal mapping, an algorithmic procedure (involving the proximal mapping which provides a gradient descent), then called an proximal algorithm. For example, gradient descent, gradient flow, alternating directions MM, etc.

4.2.2 Fessler's model-based X-ray CT image reconstruction

Use duality and group coordinate ascent (not descent because of dual variables). Additionally employ parallelism and maximise throuhgput via interleaving memory- and computationally-intensive tasks.

Consider typical imaging problem:

$$\hat{x} = \arg\min_{x} L(Ax) + R(Cx) + I(x) , \tag{4.8}$$

where L is the data-fidelity term with system matrix A, R is a regulariser with finite differences C, and I accounts for non-negativity.

L: large, slow gradients, poorly conditioned, but highly correlated views (ordered subsets)

R: non-quadratic, sometimes non-differentiable, but localised Markov-like

I: hard for line searches, easy clamping

Consider an iterative procedure

$$x_{n+1} = \arg\min_{x} \frac{\mu}{2}\|x - x_n\|_2^2 + L(Ax) + R(Cx) + I(x) , \tag{4.9}$$

Rewrite 4.9 in terms of dual variables u, v, and z using convex conjugates

$$x_{n+1} = \arg\min_{x} \sup_{u,v,z} S_n(x, u, v, z) \tag{4.10}$$

$$S_n(x, u, v, z) = \frac{\mu}{2}\|x - x_n\|_2^2 + x^T(A^T u + C^T v + z) - L^*(u) - R^*(v) - I^*(z) . \tag{4.11}$$

The duality approach thus reads:

$$D_n(u, v, z) = -\frac{1}{2\mu}\|(A^T u + C^T v + z)\|_2^2 - L^*(u) - R^*(v) - I^*(z) . \tag{4.12}$$

and

$$u_{n+1}, v_{n+1}, z_{n+1} \approx \arg\max_{u,v,z} D_n(u, v, z) \tag{4.13}$$

$$x_{n1+1} = x_n - \frac{1}{\mu}(A^T u_{n+1} + C^T v_{n+1} + z_{n+1}) . \tag{4.14}$$

Apply group coordinate ascent in each iteration

4.2.3 Chambolle's dual method

[Cha04] Based on a dual formulation and related to Chan, Golub, and Mulet [CGM99]. Convergence proofed. For TV regularisation, the non-smooth primal objective is transformed into a simple, quadratic, thus smooth, dual objective, but now with additional quadratic constraints. The resulting dual optimisation can be addressed using the Karush-Kuhn-Tucker (KKT) optimality system.

4.2.4 Primal-dual hybrid gradient method

primal-dual hybrid gradient (PDHG) methods alternates between primal and dual formulation. Saddle-point formulation:

$$\inf_{u} \sup_{|\mathbf{p}|\leq 1} G(u, \mathbf{p}) \quad \text{with} \quad G(u, \mathbf{p}) := \frac{1}{2}\int_{\Omega} (Ku - f)^2 \, \mathbf{dx} + \lambda \int_{\Omega} u \, \text{div}\mathbf{p} \, \mathbf{dx} \tag{4.15}$$

Fixing the primal or the dual variable, results in two subproblems

$$\sup_{|\mathbf{p}|\leq 1} G(u, \mathbf{p}) \quad \text{and} \quad \inf_{u} G(u, \mathbf{p}) \,. \tag{4.16}$$

Rationale: If neither the primal nor the dual variable is optimal, there is little to gain by iterating each subproblem until convergence. Thus, one step of gradient ascent/descent is applied to each subproblem alternatively.

The PDHG method of Zhu and Chan is faster than Bregman iteration which is faster than Chambolle's semi-implicit dual method.

4.2.5 Semi-smooth Newton method

Rewrite KKT optimality conditions by combining complementarity and Lagrange multipliers into single equality constraint using the Fischer-Burmeister function $\phi(a, b) := \sqrt{a^2 + b^2} - a - b$. The KKT system then reads

$$\begin{aligned} \mu\mathbf{p} &= H(\mathbf{p}) \\ \phi(\mu, 1 - |\mathbf{p}|) &= 0 \,, \end{aligned} \tag{4.17}$$

which is semi-smooth. Solve e. g. using semi-smooth Newton's method.

For the dual problem in 4.2.3, Newton's method may be singular.

4.3 Total variation

4.3.1 Theory

Introduced by Rudin, Osher, and Fatemi [ROF92] as a regularising criterion for solving inverse problems. Total-variation (TV)-based image restoration models according to Rudin, Osher, Fatemi are dubbed ROF. Example of variational partial differential equaiton (PDE)-based edge-preserving denoising models. Unconstrained TV reads

$$\inf_{u \in L^2(\Omega)} \int_{\Omega} |\nabla u| + \mu \int_{\Omega} (u - f)^2 \mathbf{dx} \tag{4.18}$$

4.3.2 Application to phantom data

Kati:

3D TV: α = 1e-1 (ad hoc method)

β = 1e-4 (is absolute value is approximated by $|f| = \sqrt{f^2 + \beta}$

2D TV: α = 1472 (determined by the S-curve). No smoothing/approximation of the TV. So far I have only applied S-curve in 2D cases

5 Various

Thomas Köhler, Hamburg. iter.reg

MeVisLab

Medical Image Processing and Visualization

MeVisLab represents a powerful modular framework for image processing research and development with a special focus on medical imaging. It allows fast integration and testing of new algorithms and the development of clinical application prototypes.

MeVisLab includes advanced software modules for segmentation, registration, volumetry, as well as quantitative morphological and functional analysis. Several clinical prototypes have been realized on the basis of MeVisLab, including software assistants for neuro-imaging, dynamic image analysis, surgery planning, and cardiovascular analysis.

The implementation of MeVisLab makes use of a number of well known third-party libraries and technologies, most importantly the application framework Qt, the visualization and interaction toolkit Open Inventor, the scripting language Python, and the graphics standard OpenGL.

MeVisLab is developed by MeVis Medical Solutions AG in close cooperation with the research institute Fraunhofer MEVIS. A part of the modules contained in the MeVisLab distribution are directly contributed by Fraunhofer MEVIS.

Osirix, Mia Plugin, Mia Lite: www.mia-solution.com. Ultra-fast level set based segmentation in 3D. seg: level-set (hierarchical) coherent propagation

MPI: magnetic particle imaging. Bernhard Gleich, Jürgen Weiszäck(er=

5.1 CT

low contrast detectability

axial plane resolution

coronal plane resolution

5.2 Philips Meeting

Collaboration with KI

5.2.1 Spectral CT

iterative reconstruction

Decomposition of Compton scattering and photo-absorption:

Radon -> l_{E_1}: decomposition -> Compton -> iteration: μ_C

Radon -> l_{E_2}: decomposition -> Photo-absorption -> iteration: μ_p
Spectral modalities: single, multi, fully
image: 512 x 700
$N_{\mathrm{proj}} \sim \sqrt{N_{\mathrm{image}}}$
SPS or sperable quadratic surrogate (SQS) algorithms
Time constraints: 10 min for whole problem
Used graphics card Nvidia K10
from single to multi: dramatic slowdown of convergence
positiviy constraint problematic: introduces bios for near zero values

5.2.2 Joint reconstruction

Denoising TV l_p-norm

6 Future development and plans

Implement variants of Chambolle-Pock including TV-regularisation to identify piecewise constant brain structure (GM, WM, CSF) and positivity constraint. Also try l_1 data-error norm representing a robust fit to the data. The idea of robust approximation is to only weakly penalise data outliers and drives small data errors towards zeros [**BoydVandenberghe**]. Moreover, try entropy-type Kullback-Leibler divergence in combination with TV.

Dictionary based approaches. Use MRI and/or high-dose CT to create and train a dictionary.

ASTRA All Scale Tomographic Reconstruction Antwerp
CPU central processing unit
CSF cerebrospinal fluid
GATE GATE
GM grey matter
GPU graphics processing unit Insight Segmentation and Registration Toolkit
KKT Karush-Kuhn-Tucker
MRI magnetic resonance imaging
NVIDIA NVIDIA
NiftyRec NiftyRec
PDHG primal-dual hybrid gradient
PyASTRA python interface to the ASTRA toolbox
ODL operator disrectisation library
ODL operator disrectisation library
PDE partial differential equaiton Reconstruction Toolkit
SPAMS SPArse Modeling Software
TomoPy Tomographic Reconstruction in Python
WM white matter