Optimising Big Images

Tuomo Valkonen

Abstract We take a look at big data challenges in image processing. Real-life photographs and other images, such ones from medical imaging modalities, consist of tens of million data points. Mathematically based models for their improvement – due to noise, camera shake, physical and technical limitations, etc. – are moreover often highly non-smooth and increasingly often non-convex. This creates significant optimisation challenges for the application of the models in quasi-real-time software packages, as opposed to more ad hoc approaches whose reliability is not as easily proven as that of mathematically based variational models. After introducing a general framework for mathematical image processing, we take a look at the current state-of-the-art in optimisation methods for solving such problems, and discuss future possibilities and challenges.

1 Introduction: Big image processing tasks

A photograph taken with current state-of-the-art digital cameras has between 10 and 20 million pixels. Some cameras, such as the semi-prototypical Nokia 808 PureView have up to 41 million sensor pixels. Despite advances in sensor and optical technology, technically perfect photographs are still elusive in demanding conditions – although some of the more artistic inclination might say that current cameras are already too perfect, and opt for the vintage. With this in mind, in low light even the best cameras however produce noisy images. Casual photographers also cannot always hold the camera steady, and the photograph becomes blurry despite advanced shake reduction technologies. We are thus presented with the challenge of improving the photographs in post-processing. This would desirably be an automated process, based on mathematically well understood models that can be relied

Tuomo Valkonen
Department of Applied Mathematics and Theoretical Physics, University of Cambrige, UK, e-mail: tuomo.valkonen@iki.fi

upon to not introduce undesired artefacts, and to restore desired features as well as possible.

The difficulty with real photographs of tens of millions of pixels is that the resulting optimisation problems are huge, and computationally very intensive. Moreover, state-of-the-art image processing techniques generally involve non-smooth *regularisers* for the modelling of our prior assumptions of what a good photograph or image looks like. This causes further difficulties in the application of conventional optimisation methods. State-of-the-art image processing techniques based on mathematical principles are only up to processing tiny images in real time. Further, choosing the right parameters for simple Tikhonov regularisation models can be difficult. Parametrisation can be facilitated by computationally more difficult iterative regularisation models [67] with easier parametrisation, or through parameter learning [84, 74, 85, 54]. These processes are computationally very intensive, requiring processing the data for multiple parameters in order to find the optimal one. The question now is, can we design fast optimisation algorithms that would make this and other image processing tasks tractable for real high-resolution photographs?

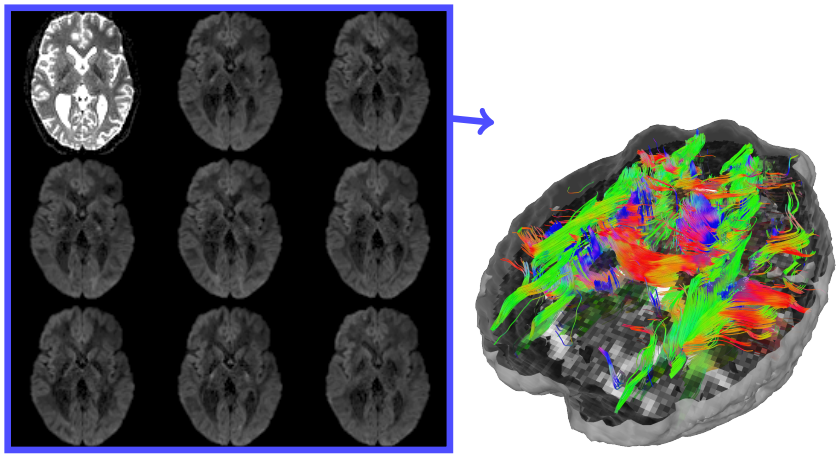

Fig. 1 Illustration of diffusion tensor imaging and tractography process. Multiple diffusion-weighted MRI images with different diffusion-sensitising gradients are first taken (left). After processing a tensor field that can be produced from these, neural pathways can be discovered through the tractography process (right). (*The author would like to thank Karl Koschutnig for the raw data, and Kristian Bredies for producing with DSI Studio the tractography image with from the diffusion tensor image computed by the author.*)

Besides photography, big image processing problems can be found in various scientific and medical areas, such magnetic resonance imaging (MRI). An example is full three-dimensional diffusion tensor MRI, and the discovery of neural pathways, as illustrated in Figure 1. I will not go into physical details about MRI here, as the focus of the chapter is in general-purpose image processing algorithms, not in particular applications and modelling. It suffices to say that diffusion tensor imag-

ing [127] combines multiple diffusion weighted MRI images (DWI) into a single tensor image u. At each point, the tensor $u(x)$ is the 3×3 covariance matrix of a Gaussian probability distribution for the diffusion direction of water molecules at x. Current MRI technology does not have nearly as high resolution as digital cameras; a $256 \times 256 \times 64$ volume would be considered to have high resolution by today's standards. However, each tensor $u(x)$ has six elements. Combined this gives 25 million variables. Moreover, higher-order regularisers such as TGV^2 [16], which we will discuss in detail in Section 3, demand additional variables for their realisation; using the PDHGM (Chambolle-Pock) method, one requires 42 variables per voxel x [135], bringing the total count to 176 million variables. Considering that a double precision floating point number takes eight bytes of computer memory, this means 1.4 gigabytes of variables. If the resolution of MRI technology can be improved, as would be desirable from a compressed sensing point of view [1], the computational challenges will grow even greater.

Due to sparsity, modelled by the geometric regularisers, imaging problems have structure that sets them apart from general big data problems. This is especially the case in a compressed sensing setting. Looking to reconstruct an image from, let's say, partial Fourier samples, there is actually very little *source data*. But the solution that we are looking for is *big*, yet, in a sense, *sparse*. This, and the poor separability of the problems, create a demand for specialised algorithms and approaches. Further big data challenges in imaging are created by convex relaxation approaches that seek to find global solutions to non-convex problems by solving a relaxed convex problem in a bigger space [34, 108, 109, 110, 112, 77]. We discuss such approaches in more detail in Section 7.1.

Overall, in this chapter, we review the state of the art of optimisation methods applicable to typical image processing tasks. The latter we will discuss shortly. In the following two sections, Section 2 and Section 3, we then review the typical mathematical *regularisation of inverse problems* approach to solving such imaging problems. After this, we look at optimisation methods amenable to solving the resulting computational models. More specifically, in Section 4 we take a look at first-order methods popular in the mathematical imaging community. In Section 5 we look at suggested second-order methods, and in Section 6 we discuss approaches for the two related topics of problems non-linear forward operators, and iterative regularisation. We finish the chapter in Section 7 with a look at early research into handling larger pictures through decomposition and preconditioning techniques, as well as the big data challenges posed by turning small problems into big ones through convex relaxation.

1.1 Types of image processing tasks

What kind of image processing tasks there are? At least mathematically the most basic one is the one we began with, *denoising*, or removing noise from an image. For an example, see Figure 1.1. In photography, noise is typically the result of low

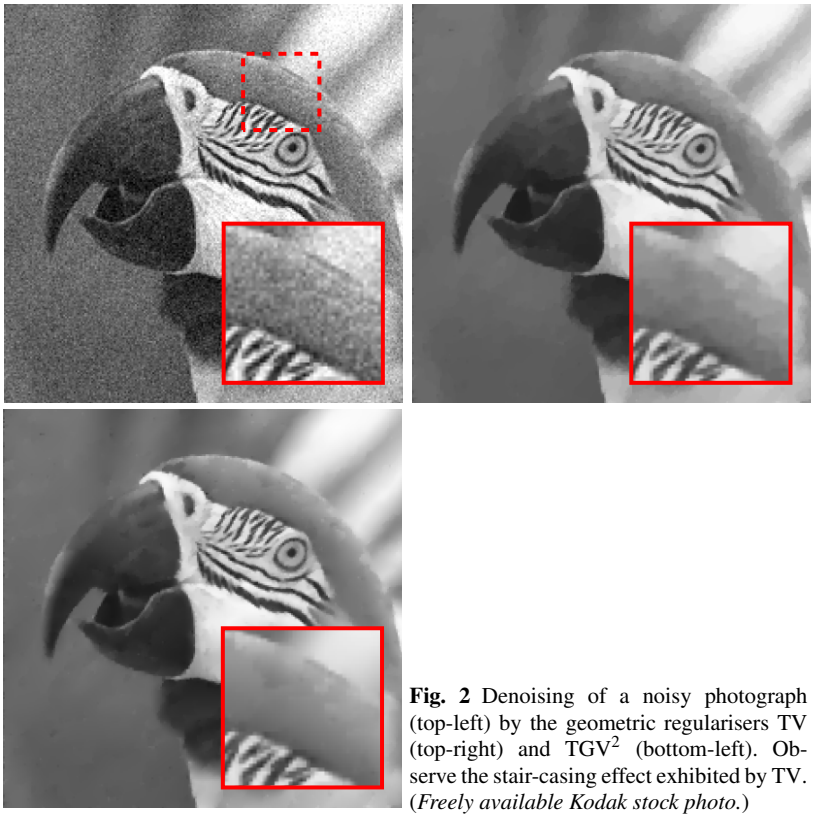

Fig. 2 Denoising of a noisy photograph (top-left) by the geometric regularisers TV (top-right) and TGV^2 (bottom-left). Observe the stair-casing effect exhibited by TV. (*Freely available Kodak stock photo.*)

light conditions, which in modern digital cameras causes the CCD (charge-coupled device) sensor array of the camera to not be excited enough. As a result the sensor images the electric current passing through it. Within the context of photography, another important task is *deblurring* or *deconvolution*, see Figure 3. Here one seeks to create a sharp image out of an unsharp image, which might be the result of camera shake – something that can also be avoided to some extent in sufficient light conditions by mechanical shake reduction technologies. In *dehazing* one seeks to remove translucent objects – clouds, haze – that obscure parts of an image and make it unsharp; see [46] for an approach fitting our variational image processing framework and further references.

Another basic task is *regularised inversion*. This involves the computation of an image from data in a different domain, such as the frequency domain. When only partial data is available, we need to add additional information into the problem in terms of the aforementioned regularisers. Problems of this form can be found in many medical imaging modalities such as magnetic resonance imaging (MRI, [135, 11, 69, 70]), positron emission tomography (PET, [141, 119]), electrical impedance tomography (EIT, [90]), computed tomography (CT, [101]), and diffuse optical tomography (DOT, [5, 71]) – the references providing merely a few

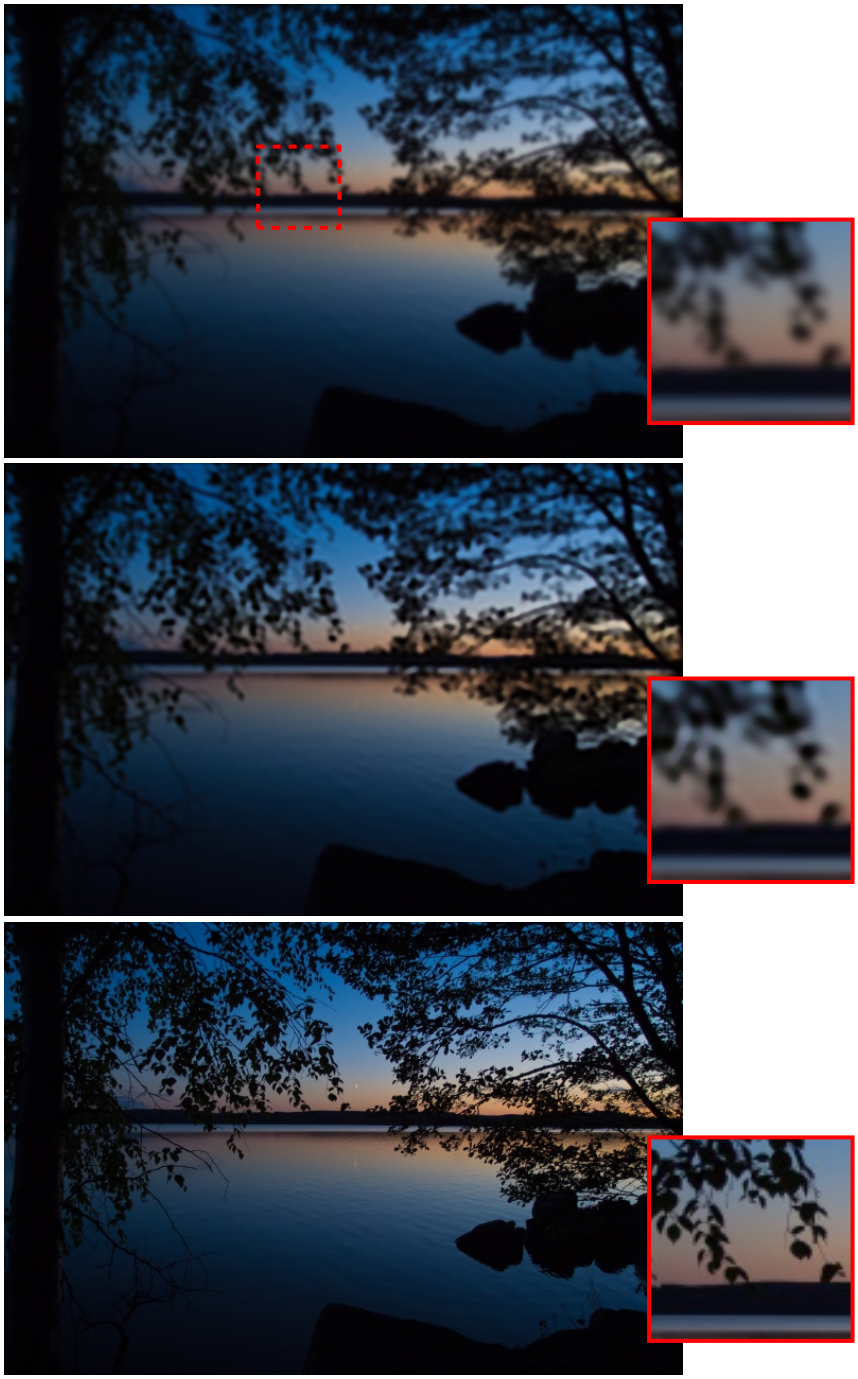

Fig. 3 Deblurring example. Could we do better, and faster? (top) Lousy photograph. (middle) TGV^2 deblurring. $\min_u \frac{1}{2}\|f-\rho_\varepsilon * u\| + \mathrm{TGV}^2(u)$ for blur kernel ρ_ε. (bottom) Ideal photograph. (*Author's own photograph. All rights withheld. Reprinted with permission.*)

starting points. Related imaging modalities can be found in the earth and planetary sciences, for example seismic tomography [83] and synthetic aperture radar (SAR, [33]). In *image fusion* one seeks to combine the data from many such modalities in order to obtain a better overall picture [15, 123, 68, 42]. In many computer vision tasks, including the automated understanding of medical imaging data, a task of paramount importance is *segmentation* [138, 91, 34, 4, 109]. Here, we seek to differentiate or detect objects in an image in order understand its content by higher-level algorithms. This may also involve *tracking* of the objects [143, 93], and in turn provides a connection to video processing, and tasks such as *optical flow* computation [65, 6, 30, 31, 130].

2 Regularisation of inverse problems

We consider image processing tasks as *inverse problems* whose basic setup is as follows. We are presented with data or measurements f, and a *forward operator* A that produced the data f, possibly corrupted by noise ν, from an unknown $\hat{u}$ that we wish to recover. Formally $f = A\hat{u} + \nu$. In imaging problems $\hat{u}$ is the uncorrupted ideal image that we want, and f the corrupted, transformed, or partial image that we have. The operator A would be the identity for denoising, a convolution operator for deblurring, and a (sub-sampled) Fourier, Radon, or other transform operator for regularised inversion. Besides the noise ν, the difficulty in recovering $\hat{u}$ is that the operator A is ill-conditioned, or simply not invertible. The overall problem is ill-posed. We therefore seek to add some prior information to the problem, to make the it well-posed. This comes in terms of a regularisation functional R, which should model our domain-specific prior assumptions of what the solution should look like. Modelling the noise and the operator equation by a fidelity functional G, the *Tikhonov regularisation* approach then seeks to solve

$$\min_u \; G(u) + \alpha R(u) \tag{P_α}$$

for some *regularisation parameter* $\alpha > 0$ that needs to be determined. Its role is to balance between regularity and good fit to data. If the noise ν is Gaussian, as is often assumed, we take

$$G(u) := \frac{1}{2}\|f - Au\|_2^2. \tag{1}$$

The choice of the regulariser R depends heavily on the problem in question; we will shortly discuss typical and emerging choices of R for imaging problems.

A major difficulty with the Tikhonov approach (P_α) is that the regularisation parameter α is difficult to choose. Moreover, with the L^2-squared fidelity (1), the scheme suffers from loss of contrast, as illustrated in [11]. If the noise level $\bar{\sigma}$ is known, an alternative approach is to solve the constrained problem

$$\min_u \; R(u) \quad \text{subject to} \quad G(u) \leq \bar{\sigma}. \tag{$\mathrm{P}^{\bar{\sigma}}$}$$

Computationally this problem tends to be much more difficult than (P_α). An approach to *estimate* solutions to this is provided by *iterative regularisation* [43, 67, 120], which we discuss in more detail in Section 6.2. The basic idea is to take a suitably chosen sequence $\alpha_k \searrow 0$. Letting $k \to \infty$, one solves (P_α) for $\alpha = \alpha_k$ to obtain u^k, and stops when $F(u^k) \leq \bar{\sigma}$. This stopping criterion is known as Morozov's *discrepancy principle* [89]. Various other a priori and a posteriori heuristics also exist. Besides iterative regularisation and heuristic stopping rules, another option for facilitating the choice of α is computationally intensive parameter learning strategies [84, 74, 85, 54], which can deal with more complicated noise models as well.

3 Non-smooth geometric regularisers for imaging

The regulariser R should try to restore and enhance desired image features without introducing artefacts. Typical images feature smooth parts as well as non-smooth geometric features such as edges. The first "geometric regularisation" models in this context have been proposed in the pioneering works of Rudin-Osher-Fatemi [118] and Perona-Malik [106]. In the former, total variation (TV) has been proposed as a regulariser for image denoising, that is $R(u) = \mathrm{TV}(u)$. Slightly cutting corners around distributional intricacies, this can be defined as the one-norm of the gradient. The interested reader may delve into all the details by grabbing a copy of [3]. In the typical case of *isotropic* TV that does not favour any particular directions, the pointwise or pixelwise base norm is the two-norm, so that

$$\mathrm{TV}(u) := \|\nabla u\|_{2,1} := \int_\Omega \|\nabla u(x)\|_2 \, dx$$

The Rudin-Osher-Fatemi (ROF) model is then

$$\min_u \frac{1}{2}\|f-u\|_2^2 + \alpha \mathrm{TV}(u), \tag{2}$$

where $u \in L^1(\Omega)$ is our unknown image, represented by a function from the domain $\Omega \subset \mathbb{R}^n$ into intensities in $\mathbb{R}$. Typically Ω is a rectangle in $\mathbb{R}^2$ or a cube in $\mathbb{R}^3$, and its elements represent different points or coordinates $x = (x_1, \ldots, x_n)$ within the n-dimensional image. For simplicity we limit ourselves in this introductory exposition to greyscale images with intensities in $\mathbb{R}$. With $\mathscr{D} := C_c^\infty(\Omega; \mathbb{R}^n)$, the total variation may also be written

$$\mathrm{TV}(u) = \sup\left\{ \int_\Omega \nabla^* \phi(x) u(x) \, dx \,\middle|\, \phi \in \mathscr{D}, \sup_{x\in\Omega} \|\phi(x)\|_2 \leq 1 \right\}, \tag{3}$$

which is useful for primal-dual and predual algorithms. Here $\nabla^* = -\operatorname{div}$ is the conjugate of the gradient operator.

The ROF model (2) successfully eliminates Gaussian noise and at the same time preserves characteristic image features like edges and cartoon-like parts. It however

has several shortcomings. A major one is the staircasing effect, resulting in blocky images; cf. Figure 1.1. It also does not deal with texture very well. Something better is therefore needed. In parts of the image processing community coming more from the engineering side, the BM3D block-matching filter [36] is often seen as the state-of-the-art method for image denoising specifically. From the visual point of view, it indeed performs very well with regard to texture under low noise levels. Not based on a compact mathematical model, such as those considered here, it is however very challenging to analyse, to prove its reliability. It, in fact, appears to completely break down under high noise, introducing very intrusive artefacts [45]. In other parts of the image and signal processing community, particularly in the context of compressed sensing, promoting sparsity in a wavelet basis is popular. This would correspond to a regulariser like $R(u) = \|Wu\|_1$, for W a wavelet transform. The simplest approaches in this category also suffer from serious artefacts, cf. [124, page 229] and [11].

To overcome some of these issues, second- (and higher-) order geometric regularisers have been proposed in the last few years. The idea is to intelligently balance between features at different scales or orders, correctly restoring all three, smooth features, geometric features such as edges, and finer details. Starting with [87], proposed variants include total generalised variation (TGV, [16]), infimal convolution TV (ICTV, [25]), and many others [22, 103, 27, 37, 39]. Curvature based regularisers such as Euler's elastica [29, 122] and [12] have also recently received attention for the better modelling of curvature in images. Further, non-convex total variation schemes have been studied in the last few years for their better modelling of real image gradient distributions [66, 60, 61, 99, 63], see Figure 4. In the other direction, in order to model texture in images, "lower-order schemes" have recently been proposed, including Meyer's G-norm [88, 139] and the Kantorovich-Rubinstein discrepancy [76]. (Other ways to model texture include non-local filtering schemes such as BM3D and NL-means [21, 36].) These models have in common that they are generally non-smooth, and increasingly often non-convex, creating various optimisation challenges. The Mumford-Shah functional [91] in particular, useful as a choice of R for segmentation, is computationally extremely difficult. As a result, either various approximations [4, 138, 109] or convex relaxation techniques are usually employed. We will will take a brief look at the latter in Section 7.1.

Due to its increasing popularity, simplicity, and reasonably good visual performance, we concentrate here on second-order *total generalised variation* (TGV [16], pun intended) as our example higher-order regulariser. In the *differentiation cascade form* [20, 17], it may be written for two parameters $(\beta,\alpha) > 0$ as

$$\mathrm{TGV}^2_{(\beta,\alpha)}(u) := \min_{w\in L^1(\Omega;\mathbb{R}^n)} \alpha\|\nabla u - w\|_{2,1} + \beta\|\mathscr{E}w\|_{F,1}.$$

Here $\mathscr{E}w$ is the symmetrised gradient, defined as

$$\mathscr{E}w(x) := \frac{1}{2}(\nabla u(x) + [\nabla u(x)]^T) \in \mathbb{R}^{n\times n}.$$

The norm in

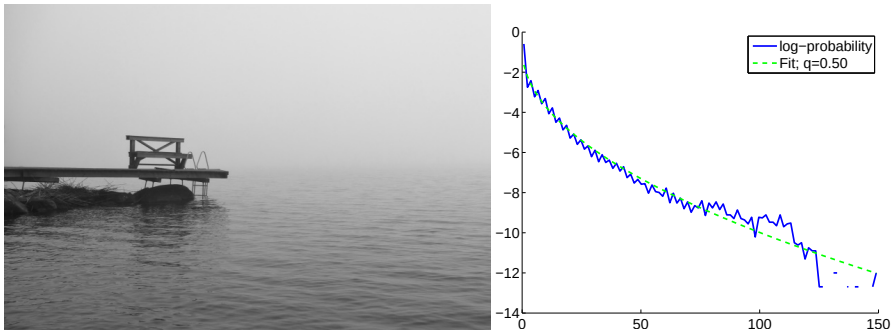

Fig. 4 Illustration of image gradient statistics. (left) Original image. (right) Log-probability (vertical axis) of gradient magnitude (horizontal axis) and optimal $t \mapsto \alpha t^q$ model fit. The optimal $q = 0.5$ causes $R(u) = \int_\Omega \|\nabla u(x)\|^q \, dx$ to become non-convex. (*Author's own photograph. All rights withheld. Reprinted with permission.*)

$$\|\mathscr{E}w\|_{F,1} := \int_\Omega \|\mathscr{E}w(x)\|_F \, dx,$$

is based on the pointwise Frobenius norm, which makes the regulariser rotationally invariant [135]. Again we slightly cut corners with distributional intricacies.

The idea in TGV^2 is that the extra variable w, over which we minimise, extracts features from u that are rather seen as second-order features. The division between first-order and second-order features is decided by the ratio β/α. If β is very large, TGV^2 essentially becomes TV, i.e., a first-order regulariser, while if β is small, all features of order larger than zero are *gratis*. In other words, only singularities, such as edges, are penalised. The use of the symmetrised gradient demands an explanation. A rationale is that if $w = \nabla v$ is already the gradient of a smooth function v, then $\nabla w = \nabla^2 v$ is symmetric. This connects TGV^2 to ICTV, which can be formulated as

$$\mathrm{ICTV}_{(\beta,\alpha)}(u) := \min_{v \in L^1(\Omega)} \alpha \|\nabla u - \nabla v\|_{2,1} + \beta \|\nabla^2 v\|_{F,1}.$$

Indeed,

$$\mathrm{TGV}^2_{(\beta,\alpha)}(u) \leq \mathrm{ICTV}_{(\beta,\alpha)}(u) \leq \alpha \mathrm{TV}(u),$$

so that TGV^2 penalises higher-order features less than ICTV or TV.

A simple comparison of TGV^2 versus TV, showing how it avoids the stair-casing effect of TV, can be found in Figure 1.1. While quite a bit is known analytically about the artefacts introduced and features restored by TV [114, 23, 28, 41], a similar study of TGV^2 and other advanced regularisers is a challenging ongoing effort [17, 134, 131, 102, 104, 86]. A more complete analytical understanding would be desirable towards the reliability of any regularisation method in critical real-life applications.

4 First-order optimisation methods for imaging

Popular, well-performing, optimisation methods in the imaging community tend to be based on variations of operator splitting and proximal (backward) steps. These include the primal-dual method of Chambolle-Pock(-Bischof-Cremers) [26, 109], the alternating directions method of multipliers (ADMM) and other Augmented Lagrangian schemes [50], as well as FISTA [9, 8, 82]. While asymptotic convergence properties of these methods are, in general, comparable to the gradient descent method, in special cases they reach the $O(1/N^2)$ rate of Nesterov's optimal gradient method [94]. Folklore also tells us that they tend to reach a visually acceptable solution in fewer iterations. The performance of the methods unfortunately decreases as the problems become increasingly ill-conditioned [81].

In all of the methods of this section, it is crucial to be able to calculate a proximal map, which we will shortly introduce. We gradually move from methods potentially involving difficult proximal maps to ones that ease or partially eliminate their computation. Generally the ones with difficult proximal maps are more efficient, if the map can be computed efficiently. We first look at primal methods, especially FISTA and its application to TV denoising in Section 4.2. We then study primal-dual methods in Section 4.3, concentrating on the PDHGM, and Section 4.4, where we concentrate on the GIST. First we begin with a few remarks about notation and discretisation, however.

4.1 Remarks about notation and discretisation

Remark 1 (Discretisation). The methods considered in this section are in principle for finite-dimensional problems, and stated in this way. We therefore have to discretise our ideal infinite-dimensional models in Section 3. We take a cell width $h > 0$, and set

$$\Omega_h := h\mathbb{Z}^n \cap \Omega.$$

Then, if $u : \Omega_h \to \mathbb{R}$, we define

$$\mathrm{TV}(u) := \sum_{x \in \Omega_h} h^n \|\nabla_h u(x)\|_2,$$

for ∇_h a suitable discrete gradient operator on Ω_h, e.g., a forward-differences operator. Similarly to the dual form (3), we also have

$$\mathrm{TV}(u) = \sup \left\{ \sum_{x \in \Omega_h} h^n \nabla^* \phi(x) u(x) \middle| \phi \in \mathscr{D}_h, \sup_{x \in \Omega_h} \|\phi(x)\|_2 \leq 1 \right\},$$

where $\mathscr{D}_h$ denotes the set of all functions $\phi : \Omega_h \to \mathbb{R}$. Likewise, we replace the operator $A : L^1(\Omega) \to Z$ in (1), for any given space $Z \ni f$, by a discretisation A_h,

discretising Z if necessary. Often in regularised inversion, Z is already discrete, however, as we have a finite number of measurements $f = (f_1, \ldots, f_m)$.

In the following, we generally drop the subscript h and, working on an abstract level, making no distinction between the finite-dimensional discretisations, and the ideal infinite-dimensional formulations. The algorithms will always be applied to the discretisations.

Remark 2 (Notation). In the literature more on the optimisation than imaging side, often the primal unknown that we denote by u is denoted x, and the dual unknown that we denote by p is denoted by y. We have chosen to use u for the unknown image, common in the imaging literature, with x standing for a coordinate, i.e., the location of a pixel within an image. Likewise, sometimes the role of the operators A and K are interchanged, as is the role of the functionals F and G. With regard to these, we use the convention in [26]. K is then always an operator occurring in the saddle-point problem ($\mathrm{P}_{\mathrm{saddle}}$), and A occurs within the functional G, as in (1). These notations are exactly the opposite in [82].

The spaces X and Y are always suitable finite-dimensional Hilbert spaces (isomorphic to $\mathbb{R}^k$ for some k), usually resulting from discretisations of our ideal infinite-dimensional image space and its predual.

4.2 Primal: FISTA, NESTA, etc.

Perhaps the best-known primal method for imaging problems is FISTA, or the *Fast Iterative Shrinkage-Thresholding Algorithm* [9]. It is based on the *forward-backward splitting* algorithm [79, 105]. A special case of this method has been derived in the literature multiple times through various different means – we refer to [38] for just one such derivation – and called the Iterative Shrinkage-Thresholding Algorithm or ISTA. FISTA adds to this an acceleration scheme similar to Nesterov's optimal gradient method [94]. The method solves a general problem of the form

$$\min_{u \in X} G(u) + F(u), \tag{$\mathrm{P}_{\mathrm{primal}}$}$$

where X is a finite-dimensional Hilbert space, e.g., a discretisation of our image space $L^1(\Omega)$. The functional $F : X \to (-\infty, \infty]$ is convex but possibly non-smooth, and $G : X \to \mathbb{R}$ is continuous with a Lipschitz continuous gradient. It is naturally assumed that the problem ($\mathrm{P}_{\mathrm{primal}}$) has a solution.

We describe FISTA in Algorithm 1. A basic ingredient of the method is the *proximal map* or *resolvent* $P_{F,\tau}$ of F. This may for a parameter $\tau > 0$, be written as

$$P_{F,\tau}(u') := \arg\min_{u} \left\{ F(u) + \frac{1}{2\tau} \|u - u'\|_2^2 \right\}.$$

Alternatively

$$P_{F,\tau}(u') = (I + \tau \partial F)^{-1}(u'),$$

Algorithm 1 FISTA [9] for ($\mathrm{P}_{\mathrm{primal}}$)

Require: L_f Lipschitz constant of ∇f.

1: Initialise $v^1 = u^0 \in X$, $t_1 := 1$, and $\tau := 1/L_f$. Set $k := 1$.
2: **repeat**
3: Compute $u^k := P_{F,\tau}(v^k - \tau \nabla G(v^k))$,
4: $t_{k+1} := \dfrac{1+\sqrt{1+4t_k^2}}{2}$,
5: $v^{k+1} := u^k + \dfrac{t_k - 1}{t_{k+1}}(u^k - u^{k-1})$.
6: Update $k := k+1$.
7: **until** A stopping criterion is fulfilled.

for ∂F the subgradient of F in terms of convex analysis; for details we refer to [115, 64]. More information about proximal maps may be found, in particular, in [117]. The update

$$u^{k+1} := P_{F,\tau}(u^k)$$

with step length τ is known as the *backward* or *proximal step*. Roughly, the idea in FISTA is to take a gradient step with respect to G, and a proximal step with respect to F. This is done in Step 3 of Algorithm 1. However, the gradient step does not use the main iterate sequence $\{u^k\}_{k=1}^\infty$, but an alternative sequence $\{v^k\}_{k=1}^\infty$, which is needed for the fast convergence. Step 4 and Step 5 are about acceleration. Step 4 changes the step length parameter for the additional sequence $\{v^k\}$, while Step 5 updates it such that it stays close to the main sequence; indeed v^{k+1} is an over-relaxed or *inertia* version of u^k – a physical interpretation is a heavy ball rolling down a hill not getting stuck in local plateaus thanks to its inertia. The sequence $t_k \to \infty$, so that eventually

$$v^{k+1} \approx 2u^k - u^{k-1}.$$

In this way, by using two different sequences, some level of second order information can be seen to be encoded into the first-order algorithm.

FISTA is very similar to Nesterov's optimal gradient method [94, 95], however somewhat simplified and in principle applicable to a wider class of functions. Step 3 is exactly the same, and the only difference is in the construction of the sequence $\{v^{k+1}\}_{k=1}^\infty$. In Nesterov's method a more general scheme that depends on a longer history is used. NESTA [10], based on Nesterov's method, is effective for some compressed sensing problems, and can also be applied to constrained total variation minimisation, that is the problem ($\mathrm{P}^{\tilde{\sigma}}$) with $R = \mathrm{TV}$ and $G(u) = \|f - Au\|_2^2$.

In principle, we could apply FISTA to the total variation denoising problem (2). We would set $G(u) = \frac{1}{2}\|f-u\|_2^2$, and $F(u) = \|\nabla u\|_1$. However, there is a problem. In order for FISTA to be practical, the proximal map $P_{\tau,F}$ has to be computationally cheap. This is not the case for the total variation seminorm. This direct approach to using FISTA is therefore not practical. The trick here is to solve the *predual problem* of (2). (In the discretised setting, it is just the dual problem.) This may be written

$$\min_{\phi\in\mathscr{D}} \frac{1}{2}\|f-\nabla^*\phi\|_2^2 \quad \text{subject to} \quad \|\phi(x)\|_2 \le \alpha \text{ for all } x\in\Omega. \tag{4}$$

We set

$$G(\phi) := \frac{1}{2}\|f-\nabla^*\phi\|_2^2, \quad \text{and} \quad F(\phi) := \delta_{B_\alpha^\infty}(\phi),$$

for

$$B_\alpha^\infty(\phi) := \{\phi\in\mathscr{D} \mid \sup_{x\in\Omega}\|\phi(x)\|_2 \le \alpha\}.$$

(We recall that for a convex set B, the indicator function $\delta_B(\phi)$ is zero if $\phi\in B$, and $+\infty$ otherwise.) Now, the proximal map $P_{F,\tau}$ is easy to calculate – it is just the pixelwise projection onto the ball $B(0,\alpha)$ in $\mathbb{R}^n$. We may therefore apply FISTA to total variation denoising [8].

One might think of using the same predual approach to solving the more difficult TGV^2 denoising problem

$$\min_u \frac{1}{2}\|f-u\|_2^2 + \mathrm{TGV}^2_{(\beta,\alpha)}(u). \tag{5}$$

The predual of this problem however has a difficult non-pointwise constraint set, and the resulting algorithm is not efficient [16]. Therefore, other approaches are needed.

4.3 Primal-dual: PDHGM, ADMM, and other variants on a theme

Both (2) and (5), as well as many more problems of the form

$$\min_u \frac{1}{2}\|f-Au\|_2^2 + R(u), \tag{6}$$

for $R=\alpha\mathrm{TV}$ or $R=\mathrm{TGV}^2_{(\beta,\alpha)}$ can in their finite-dimensional discrete forms be written as saddle-point problems

$$\min_{u\in X}\max_{p\in Y} G(u) + \langle Ku,p\rangle - F^*(p). \tag{$\mathrm{P}_{\mathrm{saddle}}$}$$

Here $G:X\to(-\infty,\infty]$ and $F^*:Y\to(-\infty,\infty]$ are convex, proper, and lower semicontinuous, and $K:X\to Y$ is a linear operator. The functional F^* is moreover assumed to be the convex conjugate of some F satisfying the same assumptions. Here the spaces X and Y are again finite-dimensional Hilbert spaces. If $P_{G_0,\tau}$ is easy to calculate for $G_0(u) := \frac{1}{2}\|f-Au\|_2^2$, then for $R=\alpha\mathrm{TV}$, we simply transform (6) into the form ($\mathrm{P}_{\mathrm{saddle}}$) by setting

$$G = G_0, \quad K = \nabla, \quad \text{and} \quad F^*(p) = \delta_{B_\alpha^\infty}(p). \tag{7}$$

For $R=\mathrm{TGV}^2_{(\beta,\alpha)}$, we write $u=(v,w)$, $p=(\phi,\psi)$, and set

Algorithm 2 PDHGM [26] for ($\mathrm{P}_{\mathrm{saddle}}$)

Require: L a bound on $\|K\|$, over-relaxation parameter θ ($\theta = 1$ usually, for convergence proofs to hold), primal and dual step lengths $\tau, \sigma > 0$ such that $\tau\sigma L^2 < 1$.

Initialise primal and dual iterate $u^1 \in X$ and $p^1 \in Y$. Set $k := 1$.
repeat
  Compute $u^{k+1} := P_{G,\tau}(u^k - \tau K^* p^k)$,
  $\bar{u}^{k+1} := u^{k+1} + \theta(u^{k+1} - u^k)$,
  $p^{k+1} := P_{F^*,\sigma}(p^k + \sigma K \bar{u}^{k+1})$.
  Update $k := k + 1$.
until A stopping criterion is fulfilled.

Algorithm 3 Accelerated PDHGM [26] for ($\mathrm{P}_{\mathrm{saddle}}$)

Require: L a bound on $\|K\|$, $\gamma > 0$ factor of strong convexity of G or F^*, initial primal and dual step lengths $\tau_1, \sigma_1 > 0$ such that $\tau_1 \sigma_1 L^2 < 1$.

Initialise primal and dual iterate $u^1 \in X$ and $p^1 \in Y$. Set $k := 1$.
repeat
  Compute $u^{k+1} := P_{G,\tau_k}(u^k - \tau_k K^* p^k)$,
  $\theta_k := 1/\sqrt{1 + 2\gamma\tau_k}$, $\quad \tau_{k+1} := \theta_k \tau_k$, $\quad$ and $\quad \sigma_{k+1} := \sigma_k / \theta_k$,
  $\bar{u}^{k+1} := u^{k+1} + \theta_k(u^{k+1} - u^k)$,
  $p^{k+1} := P_{F^*,\sigma_{k+1}}(p^k + \sigma_{k+1} K \bar{u}^{k+1})$.
  Update $k := k + 1$.
until A stopping criterion is fulfilled.

$$G(u) = G_0(v), \quad Ku = (\nabla v - w, \mathscr{E} w), \quad \text{and} \quad F(p) = \delta_{B_\alpha^\infty}(\phi) + \delta_{B_\beta^\infty}(\psi). \tag{8}$$

Observe that G in (7) for TV is strongly convex if the nullspace $\mathscr{N}(K) = \{0\}$, but G in (8) is never strongly convex. This has important implications.

Namely, problems of the form ($\mathrm{P}_{\mathrm{saddle}}$) can be solved by the Chambolle-Pock(-Bischof-Cremers) algorithm [26, 109], also called the *modified primal dual hybrid-gradient method* (PDHGM) in [44]. In the presence of strong convexity of either F^* or G, a Nesterov acceleration scheme as in FISTA can be employed to speed up the convergence to $O(1/N^2)$. The unaccelerated variant has rate $O(1/N)$. Therefore the performance of the method for TGV^2 denoising is theoretically significantly worse than for TV. We describe the two variants of the algorithm, accelerated and unaccelerated, in detail in Algorithm 2 and Algorithm 3, respectively.

The method is based on proximal or backward steps for both the primal and dual variables. Essentially one holds u and p alternatingly fixed in ($\mathrm{P}_{\mathrm{saddle}}$), and takes a proximal step for the other. However, this scheme, known as PDHG (primal-dual hybrid gradient method, [147]), is generally not convergent. That is why the *over-relaxation* or *inertia* step $\bar{u}^{k+1} := u^{k+1} + \theta(u^{k+1} - u^k)$ for the primal variable is crucial. We also need to take $\theta = 1$ for the convergence results to hold [26].

Algorithm 4 Augmented Lagrangian method for $\min_u F(u)$ subject to $Au = f$

Require: A sequence of *penalty parameters* $\mu^k \searrow 0$, initial iterate $u^0 \in X$, and initial Lagrange multiplier $\lambda^1 \in Y$.

1: Define the *Augmented Lagrangian*

$$\mathscr{L}(u,\lambda,\mu) := F(u) + \langle \lambda, Au - f\rangle + \frac{1}{2\mu}\|Au - f\|_2^2.$$

2: **repeat**

3: Compute $u^k := \arg\min_u \mathscr{L}(u,\lambda^k;\mu^k)$ starting from u^{k-1}, and

4: $\lambda^{k+1} := \lambda^k - (Au^k - b)/\mu^k$.

5: Update $k := k+1$.

6: **until** A stopping criterion is fulfilled.

Naturally inertia step on the primal variables u could be replaced by a corresponding step on the dual variable p.

It can be shown that the PDHGM is actually a preconditioned proximal point method [56], see also [116, 133]. (This reformulation is the reason why the ordering of the steps in Algorithm 2 is different from the original one in [26].) Proximal point methods apply to general monotone inclusions, not just convex optimisation, and the inertial and splitting ideas of Algorithm 2 have been generalised to those [80].

The PDHGM is very closely related to a variety other algorithms popular in image processing. For $K = I$, the unaccelerated version of the method reduces [26] to the earlier *alternating direction method of multipliers* (ADMM, [50]), which itself is a variant of the classical *Douglas-Rachford splitting algorithm* (DRS, [40]), and an approximation of the *Augmented Lagrangian method*. The idea here is to consider the primal problem corresponding to ($\mathrm{P}_{\mathrm{saddle}}$), that is

$$\min_u \; G(u) + F(Ku).$$

Then we write this as

$$\min_{u,p} \; G(u) + F(p) \quad \text{subject to} \quad Ku = p.$$

The form of the Augmented Lagrangian method in Algorithm 4 may be applied to this. If, in the method, we perform Step 3 first with respect to u and then respect to p, keeping the other fixed, and keep the *penalty parameter* μ^k constant, we obtain the ADMM. For $K = I$, this will be just the PDHGM. For $K \neq I$, the PDHGM can be seen as a preconditioned ADMM [44].

The ADMM is further related to the *split inexact Uzawa method*, and equals on specific problems the *alternating split Bregman method*. This is again based on a proximal point method employing in $P_{G,\tau}$, instead of the standard L^2-squared distance, alternative so-called Bregman distances related to the problem at hand; see [121] for an overview. We refer to [121, 44, 26] for even further connections.

Generalising, it can be said that FISTA performs better than PDHGM when the computation of the proximal mappings it requires can be done fast [26]. The PDHGM is however often one of the best performers, and often very easy to implement thanks to the straightforward linear operations and often easy proximal maps. It can be applied to TGV^2 regularisation problems, and generally outperforms FISTA, which was still used for TGV^2 minimisation in the original TGV paper [16]. The problem is that the proximal map required by FISTA for the predual formulation of TGV^2 denoising is too difficult to compute. The primal formulation would be even more difficult, being of same form as the original problem. This limits the applicability of FISTA. But the PDHGM is also not completely without these limitations.

4.4 When the proximal mapping is difficult

In typical imaging applications, with $R = \mathrm{TV}$ or $R = \mathrm{TGV}^2$, the proximal map $P_{F^*,\sigma}$ corresponding to the regulariser is easy to calculate for PDHGM – it consists of simple pointwise projections to unit balls. But there are many situations, when the proximal map $P_{G_0,\tau}$ corresponding to the data term is unfeasible to calculate on every iteration of Algorithm 2. Of course, if the operator $A = I$ is the identity, this is a trivial linear operation. Even when $A = S\mathscr{F}$ is a sub-sampled Fourier transform, the proximal map reduces to a simple linear operation thanks to the *unitarity* $\mathscr{F}^*\mathscr{F} = \mathscr{F}\mathscr{F}^* = I$ of the Fourier transform. But what if the operator is more complicated, or, let's say

$$G_0(v) = \frac{1}{2}\|f - Av\|_2^2 + \delta_C(v),$$

for some difficult constraint set C? In a few important seemingly difficult cases, calculating the proximal map is still very feasible. This includes a pointwise positive semi-definiteness constraint on a diffusion tensor field when $A = I$ [135]. Here also a form of *unitary invariance* of the constraint set is crucial [78]. If $A \neq I$ with the positivity constraint, the proximal mapping can become very difficult. If A is a pointwise (pixelwise) operator, this can still be marginally feasible if special *small data* interior point algorithms are used for its pointwise computation [136, 132]. Nevertheless, even in this case [136], a reformulation tends to be more efficient. Namely, we can rewrite

$$G_0(v) = \sup_{\lambda} \langle Av - f, \lambda \rangle - \frac{1}{2}\|\lambda\|_2^2 + \delta_C(v).$$

Then, in case of TV regularisation, we set $p = (\phi, \lambda)$, and

$$G(u) := \delta_C(u), \quad Ku := (\nabla u, Au), \quad \text{and} \quad F^*(p) := \delta_{B_\alpha^\infty}(\phi) + \langle f, \lambda \rangle + \frac{1}{2}\|\lambda\|_2^2.$$

Algorithm 5 GIST [82] for (P_{saddle}) with (9)

1: Initialise primal and dual iterate $u^1 \in X$ and $p^1 \in Y$. Set $k := 1$.
2: **repeat**
3: Compute $\bar{u}^{k+1} := u^k + A^T(f - Au^k) - K^T p^k$,
4: $p^{k+1} := P_{F^*,1}(p^k + K\bar{u}^{k+1})$,
5: $u^{k+1} := u^k + A^T(f - Au^k) - K^T p^{k+1}$.
6: Update $k := k + 1$.
7: **until** A stopping criterion is fulfilled.

The modifications for TGV^2 regularisation are analogous. Now, if the projection into C is easy, and A and A^* can be calculated easily, as is typically the case, application of Algorithm 2 becomes feasible. The accelerated version is usually no longer applicable, as the reformulated G is not strongly convex, and F^* usually isn't.

However, there may be better approaches. One is the GIST or *Generalised Iterative Soft Thresholding* algorithm of [82], whose steps are laid out in detail in Algorithm 5. As the name implies, it is also based on the ISTA algorithm as was FISTA, and is a type of forward-backward splitting approach. It is applicable to saddle-point problems (P_{saddle}) with

$$G(u) = \frac{1}{2}\|f - Au\|^2. \tag{9}$$

In essence, the algorithm first takes a forward (gradient) step for u in the saddle-point formulation (P_{saddle}), keeping p fixed. This is only used to calculate the point where to next take a proximal step for p keeping u fixed. Then it takes a forward step for u at the new p to actually update u. In this way, also GIST has a second over-relaxation type sequence for obtaining convergence. If $\|A\| < \sqrt{2}$ and $\|K\| < 1$, then GIST converges with rate $O(1/N)$. We recall that forward-backward splitting generally has rather stronger requirements for convergence, see [128] as well as [62, 121] for an overview and relevance to image processing. Also, in comparison to the PDHGM, the calculation of the proximal map of G is avoided, and the algorithm requires less variables and memory than PDHGM with the aforementioned "dual transportation" reformulation of the problem.

5 Second-order optimisation methods for imaging

Although second-order methods are more difficult to scale to large images, and the non-smoothness of typical regularisers R causes complications, there has been a good amount of work into second-order methods for total variation regularisation, in particular for the ROF problem (2). Typically some smoothing of the problem is required. The first work in this category is [140]. There, the total variation seminorm

$\|\nabla u\|_{2,1}$ is replaced by the smoothed version

$$\widetilde{\mathrm{TV}}_\varepsilon(u) := \int_\Omega \sqrt{\|\nabla u(x)\|^2 + \varepsilon}\, dx. \tag{10}$$

Then the Newton method is applied – after discretisation, which is needed for u to live in and $\widetilde{\mathrm{TV}}_\varepsilon$ to have gradients in a "nice" space. In the following, we will discuss one further development, primarily to illustrate the issues in the application of second order method to imaging problems, not just from the point of view of big data, but also from the point of view of imaging problems. Moreover, second-order methods generally find high-precision solutions faster than first-order methods when it is feasible to apply one, and are in principle more capable of finding actual local minimisers to non-convex problems. These include non-convex total variation regularisation or inversion with non-linear forward operators.

5.1 Huber-regularisation

In recent works on second order methods, Huber-regularisation, also sometimes called Nesterov-regularisation, is more common than the smoothing of (10). This has the advantage of only distorting the one-norm of TV locally for small gradients, and has a particularly attractive form in primal-dual or (pre)dual methods. Moreover, Huber-regularisation tends to ameliorate the stair-casing effect of TV. The Huber-regularisation of the two-norm on $\mathbb{R}^n$ may for a parameter $\gamma > 0$ be written as

$$|g|_\gamma := \begin{cases} \|g\|_2 - \frac{1}{2\gamma}, & \|g\|_2 \geq 1/\gamma, \\ \frac{\gamma}{2}\|g\|_2^2, & \|g\|_2 < 1/\gamma. \end{cases} \tag{11}$$

Alternatively, in terms of convex conjugates, we have the dual formulation

$$|g|_\gamma = \max\left\{ \langle g, \xi\rangle - \frac{1}{2\gamma}\|\xi\|_2^2 \middle| \xi \in \mathbb{R}^n,\, \|\xi\|_2 \leq 1 \right\}. \tag{12}$$

In other words, the sharp corner of the graph of the two-norm is smoothed around zero – the more the smaller the parameter γ is. (Sometimes in the literature, our γ is replaced by $1/\gamma$, and so smaller value is less regularisation.) In the dual formulation, we just regularise the dual variable. This helps to avoid its oscillation. With (11), we may then define the (isotropic) Huber-regularised total variation as

$$\mathrm{TV}_\gamma(u) := \int_\Omega |\nabla u(x)|_\gamma dx.$$

5.2 A primal-dual semi-smooth Newton approach

In the infinite-dimensional setting, we add for a small parameter $\varepsilon > 0$ the penalty $\varepsilon\|\nabla u\|_2^2$ to (2), to pose it in a Hilbert space. This will cause the corresponding functional to have "easy" subdifferentials without the measure-theoretic complications of working in the Banach space of functions of bounded variation. With Huber-regularisation, (2) then becomes differentiable, or "semismooth" [111, 32]. A generalised Newton's method can be applied. We follow here the "infeasible active set" approach on the predual problem (4), developed in [59], but see also [73]. In fact, we describe here the extension in [85] for solving the more general problem

$$\min_{u\in H^1(\Omega;\mathbb{R}^N)} \varepsilon\|\nabla u\|_2^2 + \frac{1}{2}\|f - Au\|_2^2 + \sum_{j=1}^{N} \alpha_j \int_\Omega |[K_j u](x)|_\gamma \, dx, \tag{P_{SSN}}$$

where $A : H^1(\Omega;\mathbb{R}^m) \to L^2(\Omega)$, and $K_j : H^1(\Omega;\mathbb{R}^m) \to L^1(\Omega;\mathbb{R}^{m_j})$, $(j = 1,\dots,N)$, are linear operators with corresponding weights $\alpha_j > 0$. This formulation is applicable to the TGV denoising problem (5) by setting $u = (v,w)$, $Au = v$, $K_1 u = \nabla v - w$, and $K_2 u = \mathscr{E} w$. The first-order optimality conditions for (P_{SSN}) may be derived as

$$-\varepsilon\Delta u + A^*Au + \sum_{i=1}^{N} K_j^* p_j = A^* f, \tag{13a}$$

$$\max\{1/\gamma, |[K_j u](x)|_2\} p_j(x) - \alpha_j [K_j u](x) = 0, \quad (j = 1,\dots,N; x \in \Omega). \tag{13b}$$

Here (13b) corresponds pointwise for the optimality of $\xi = p_j(x)/\alpha_j$ for $g = \alpha[K_j u](x)$ and $\gamma' = \gamma/\alpha_j$ in (12). To see why this is right, it is important to observe that $\alpha|g|_\gamma = |\alpha g|_{\gamma/\alpha}$. Even in a finite-dimensional setting, although we are naturally in a Hilbert space, the further regularisation by $\varepsilon\|\nabla u\|_2^2$ is generally required to make the system matrix invertible. If we linearise (13b), solve the resulting linear system and update each variable accordingly, momentarily allowing each dual variable p_j to become infeasible, and then project back into the respective dual ball, we obtain Algorithm 6. For details of the derivation we refer to [59, 85]. Following [125], it can be shown that the method converges locally superlinearly near a point where the subdifferentials of the operator on $(u, p_1, \dots p_N)$ corresponding to (13) are non-singular. Further dampening as in [59] guarantees local superlinear convergence at any point.

Remark 3. If one wants to use a straightforward Matlab implementation of Algorithm 6 with TGV^2 and expect anything besides a computer become a lifeless brick, the system (14) has to be simplified. Indeed B is invertible, so we may solve δu from

$$B\delta u = R_1 - \sum_{j=1}^{N} K_j^* \delta p_j. \tag{15}$$

Thus we may simplify δu out of (14), and only solve for $\delta p_1, \dots, \delta p_N$ using a reduced system matrix. Finally we calculate δu from (15).

Algorithm 6 An infeasible semi-smooth Newton method for ($\mathrm{P_{SSN}}$) [59, 85]

Require: Step length $\tau > 0$.

1: Define the helper functions

$$\mathfrak{m}_j(u)(x) := \max\{1/\gamma, |[K_j u](x)|_2\}, \qquad [\mathfrak{D}(p)q](x) := p(x)q(x),$$
$$\mathfrak{N}(z)(x) := \begin{cases} 0, & |z(x)|_2 < 1/\gamma, \\ \frac{z(x)}{|z(x)|_2}, & |z(x)|_2 > 1/\gamma, \end{cases} \qquad (x \in \Omega).$$

2: Initialise primal iterate u^1 and dual iterates $(p^1, \dots, p^N)$. Set $k := 1$.

3: **repeat**

4: Solve $(\delta u, \delta p_1, \dots, \delta p_N)$ from the system

$$\begin{pmatrix} B, & K_1^* & \dots & K_N^* \\ -\alpha_1 K_1 + \mathfrak{N}(K_1 u^k)^* \mathfrak{D}(p_1) K_1 & \mathfrak{D}(\mathfrak{m}_j(u^k)) & 0 & 0 \\ \vdots & 0 & \ddots & 0 \\ -\alpha_N K_N + \mathfrak{N}(K_N u^k)^* \mathfrak{D}(p_N) K_N & 0 & 0 & \mathfrak{D}(\mathfrak{m}_N(u)) \end{pmatrix} \begin{pmatrix} \delta u \\ \delta p_1 \\ \vdots \\ \delta p_N \end{pmatrix} = R \qquad (14)$$

where

$$R := \begin{pmatrix} -Bu^k - \sum_{i=1}^N K_j^* p_j^k + A^* f \\ \alpha_1 K_1 u^k - \mathfrak{D}(\mathfrak{m}_1(u)) p_1^k \\ \vdots \\ \alpha_N K_N u^k - \mathfrak{D}(\mathfrak{m}_N(u^k)) p_N^k \end{pmatrix},$$

and

$$B := -\varepsilon \Delta + A^* A.$$

5: Update

$$(u^{k+1}, \widetilde{p}_1^{k+1}, \dots \widetilde{p}_N^{k+1}) := (u^k + \tau \delta u, p_1^k + \tau \delta p_1, p_N^k + \tau \delta p_N),$$

6: Project

$$p_j^{k+1} := \mathfrak{P}(\widetilde{p}_j^{k+1}; \alpha_j), \quad \text{where} \quad \mathfrak{P}(p; \alpha)(x) := \operatorname{sgn}(p(x)) \min\{\alpha, |p(x)|\},$$

7: Update $k := k + 1$.

8: **until** A stopping criterion is satisfied.

In [85], the algorithm is compared against the PDHGM (Algorithm 2) both for TV and TGV^2 denoising, (2) and (5), respectively. It is observed that the performance can be comparable to PDHGM for TV with images up to size about 256×256. In case of TGV^2 the method performs significantly worse due to the SSN system (14) being worse-conditioned, and the data size of TGV^2 being far larger through the additional variables w and p_2. For images in the range 512×512 the method is no longer practical on current desktop computers, so definitely not for multi-million megapixel real-life photographs.

5.3 A note on interior point methods

Application of interior point methods to (2) – which can be very easily done with CVX [53, 52] – has similar scalability problems as Algorithm 6. This is mainly due to excessive memory demands. For small problem sizes the performance can be good when high accuracy is desired – especially the commercial MOSEK solver performs very well. However, as is to be expected, the performance deteriorates quickly as problem sizes increase and the interior point formulations become too large to fit in memory [75, 107].

A way forward for second-order methods is to use preconditioning to make the system matrix better conditioned, or to split the problem into smaller pieces using domain decomposition techniques. We will discuss what early progress has been made in this area in Section 7.2.

5.4 Methods for non-convex regularisers

One reason for us introducing Algorithm 6, despite being evidently not up to the processing of big images at this stage, is that the same ideas can be used derive methods for solving non-convex total variation problems [60, 61, 63]

$$\min_u \frac{1}{2}\|f - Au\|_2^2 + \alpha \int_\Omega \psi(\|\nabla u(x)\|)\,dx. \tag{16}$$

Here $\psi : [0,\infty) \to [0,\infty)$ is a concave energy that attempts to model real gradient distributions in images, recall Figure 4. Usually $\psi(t) = t^q$ for $q \in (0,1)$ although this has significant theoretical problems [63]. Alternative, first-order, approaches include the iPiano of [98], which looks a lot like FISTA, allowing F in ($\mathrm{P}_{\mathrm{primal}}$) to be non-convex and modifying the updates a little. The PDHGM has also recently been extended to "semiconvex" F [92]; this includes (16) when the energies $\psi(t) = t^q$ are linearised for small t. No comparisons between the methods are known to the author.

6 Non-linear operators and methods for iterative regularisation

We now discuss typical and novel approaches for two closely related topics: inverse problems with non-linear forward operators A, and iterative regularisation. Overall, the workhorse algorithms in this category are much less developed than for Tikhonov regularisation with linear forward operators, in which case both data and convex terms are convex.

Algorithm 7 Gauss-Newton method for (P_α) with (17)

1: Initialise primal iterate $u^1 \in X$. Set $k := 1$.
2: **repeat**
3: Solve for $u^{k+1} := u$ the convex problem

$$\min_u \frac{1}{2}\|f - A(u^k) - \nabla A(u^k)(u - u^k)\|_2^2 + \alpha R(u). \tag{18}$$

4: Update $k := k + 1$.
5: **until** A stopping criterion is fulfilled.

6.1 Inverse problems with non-linear operators

We now let A be a non-linear operator and set

$$G(u) := \frac{1}{2}\|f - A(u)\|_2^2. \tag{17}$$

Although second-order methods in particular could in principle be applied to smoothed versions of the resulting Tikhonov problem (P_α), in inverse problems research, a classical approach in this case is the Gauss-Newton method, described in Algorithm 7. It is based on linearising A at an iterate u^k and solving the resulting convex problem at each iteration until hopeful eventual convergence. This can be very expensive, and convergence is not generally guaranteed [97], as experiments in [133] numerically confirm. However, for the realisation of the algorithm, it is not necessary that R is (semi-)smooth as with Newton type methods.

A more recent related development is the *primal-dual hybrid gradient method for non-linear operators* (NL-PDHGM, [133]), which we describe in Algorithm 8. It extends the iterations of the PDHGM (Algorithm 2) to non-linear K in the saddle-point problem ($\mathrm{P}_{\text{saddle}}$). That is, it looks for critical points of the problem

$$\min_{u\in X}\max_{p\in Y} G(u) + \langle K(u), p\rangle - F^*(p), \tag{$\mathrm{P}_{\text{nl-saddle}}$}$$

where now $K \in C^2(X;Y)$, but $G : X \to (-\infty,\infty]$ and $F^* : Y \to (-\infty,\infty]$ are still convex, proper, and lower semicontinuous. Through the reformulations we discussed in Section 4.4, it can also be applied when G is as in (17) with nonlinear A. According to the experiments in [133], the NL-PDHGM by far outperforms Gauss-Newton on example problems from magnetic resonance imaging. Moreover, the non-linear models considered improve upon the visual and PSNR performance of earlier linear models in [135, 136] for diffusion tensor imaging, and in [11] for MR velocity imaging, cf. also [145]. The method can be proved to converge locally on rather strict conditions. For one, Huber-regularisation of TV or TGV^2 is required. The second peculiar condition is that the regularisation parameter α and the noise level $\bar{\sigma}$ have to be "small". An approximate linearity condition, as with the is common with the

Algorithm 8 NL-PDHGM [133] for ($P_{\text{nl-saddle}}$)

Require: L a local bound on $\|\nabla K(u)\|$ in a neighbourhood of a solution (u^*, p^*), over-relaxation parameter θ (usually $\theta = 1$ for convergence results to hold), primal and dual step lengths $\tau, \sigma > 0$ such that $\tau\sigma L^2 < 1$.

1: Initialise primal and dual iterate $u^1 \in X$ and $p^1 \in Y$. Set $k := 1$.
2: **repeat**
3: Compute $u^{k+1} := P_{G,\tau}(u^k - \tau[\nabla K(u^k)]^* p^k)$,
4: $\bar{u}^{k+1} := u^{k+1} + \theta(u^{k+1} - u^k)$,
5: $p^{k+1} := P_{F^*,\sigma}(p^k + \sigma K(\bar{u}^{k+1}))$.
6: Update $k := k + 1$.
7: **until** A stopping criterion is fulfilled.

combination of the Gauss-Newton method with iterative regularisation, discussed next, is however not required.

6.2 Iterative regularisation

We now briefly consider solution approaches for the constrained problem ($P^{\bar{\sigma}}$), which tends to be much more difficult than the Tikhonov problem (P_α). In some special cases, as we've already mentioned, NESTA [10] can be applied. One can also apply the classical Augmented Lagrangian method [97]. If one minimises R subject to the exact constraint $Au = f$, the method has the form in Algorithm 4 with a suitable rule of decreasing the penalty parameter μ^k. The latter has roughly the same role here as α in the Tikhonov problem (P_α). Thus the Augmented Lagrangian method forms a way of iterative regularisation, if we actually stop the iterations when Morozov's discrepancy principle is violated. (In this case, we do not expect $Au = f$ to have a solution, but require $\|Au - f\| \leq \bar{\sigma}$ to have a solution.) If we fix $\mu^k \equiv 1$ the Augmented Lagrangian method then corresponds [144] to so-called Bregman iterations [100, 51] on (P_α). Another way to view this is that one keeps α in (P_α) fixed, but, iterating its solution, replaces on each iteration the distance $\frac{1}{2}\|Au - f\|_2^2$ by the *Bregman distance*

$$D_G^\lambda(u, u^{k-1}) := G(u) - G(u^{k-1}) - \langle \lambda, u - u^{k-1} \rangle,$$

for $G(u) = \frac{1}{2}\|Au - f\|_2^2$, and $\lambda \in \partial G(u^{k-1})$. This scheme has a marked contrast-enhancing effect compared to the basic Tikhonov approach.

But how about just letting $\alpha \searrow 0$ in (P_α), as we discussed in Section 2, and stopping when Morozov's discrepancy principle is violated? This is equally feasible for linear A as the Augmented Lagrangian approach. But what if A is non-linear? The Gauss-Newton approach for solving each of the inner Tikhonov problems results in this case in three nested optimisation loops: one for $\alpha_k \searrow 0$, one for solving the

non-convex problem for $\alpha = \alpha_k$, and one for solving (18). Aside from toy problems, this start to be computationally unfeasible. There is some light at the end of the tunnel however: the *Levenberg-Marquardt*, and *iteratively regularised Landweber and Gauss-Newton* (IRGN) methods [14, 67]. Similar approaches can also be devised for Bregman iterations when A is nonlinear [7].

The iteratively regularised Levenberg-Marquardt scheme [43, 55, 67] is the one most straightforward for general regularisers, including the non-smooth ones we are interested in. In the general case, a convergence theory is however lacking to the best of our knowledge, unless the scheme is Bregmanised as in [7]. Bregman distances have indeed been generally found useful for the transfer of various results from Hilbert spaces to Banach spaces [120]. Nevertheless, the Levenberg-Marquardt scheme combines the Gauss-Newton step (18) with the parameter reduction scheme $\alpha_k \searrow 0$ into a single step. It then remains to solve (18) for $\alpha = \alpha_k$ with another method, such as those discussed in Section 4. For the simple, smooth, regulariser $R(u) = \|u - u_0\|^2$, not generally relevant to imaging problems, the iteratively regularised Landweber and Gauss-Newton methods can combine even this into a single overall loop. Convergence requires, in general, a degree of *approximate linearity* from A. In the worst case, this involves the existence of $\eta, \rho > 0$ and a solution u^* of $A(u^*) = f$ such that

$$\|A(\widetilde{u}) - A(u) - \nabla A(u)(\widetilde{u} - u)\| \leq \eta \|u - \widetilde{u}\| \|A(u) - A(\widetilde{u})\|, \tag{19}$$

whenever u and $\widetilde{u}$ satisfy $\|u - u^*\|, \|\widetilde{u} - u^*\| \leq \rho$. Although (19) and related conditions can be shown to hold for certain non-linear parameter identification problems, in general it is rarely satisfied [43, 67]. For example, (19) is not satisfied by the operators considered for magnetic resonance imaging (MRI) in [133], where Algorithm 8 was developed.

7 Emerging topics

We finish this chapter with a brief overlook at a couple of topics that have the potential to improve image optimisation performance, and turn other challenges into big data challenges. The latter, discussed first, is convex relaxation, which transforms difficult non-convex problems into large-scale convex problems. The former is decomposition and preconditioning techniques, which seek to turn large problems into smaller ones.

7.1 Convex relaxation

The basic idea behind convex relaxation approaches is to lift a non-convex problem into a higher-dimensional space, where it becomes convex. This kind of approaches

are becoming popular in image processing, especially in the context of the difficult problem of segmentation, and the Mumford-Shah problem [91]. This may be written

$$\min_u \frac{1}{2}\|f-u\|_2^2 + \alpha \int_\Omega \|\nabla u(x)\|_2^2\,dx + \beta \mathscr{H}^{n-1}(J_u). \tag{20}$$

Here u is a *function of bounded variation*, which may have discontinuities J_u, corresponding to boundaries of different objects in the scene f. The final term measures their length, and the middle term forces u to be smooth outside the discontinuities; for details we refer to [3]. The connected components of Ω, as split by J_u, allow us to divide the scene into different segments.

Let us consider trying to solve for $u \in L^1(\Omega)$ and a general non-convex G the problem

$$\min_{u\in L^1(\Omega)} G(u). \tag{21}$$

Any global solution of this problem is a solution of

$$\min_{u\in L^1(\Omega)} \overline{G}(u),$$

where $\overline{G}$ is the convex lower semicontinuous envelope of G, or the greatest lower semicontinuous convex function such that $\overline{G} \le G$. The minimum values of the functionals agree, and under some conditions, the minimisers of G and $\overline{G}$ agree.

But how to compute $\overline{G}$? It turns out that in some cases [24], it is significantly easier to calculate the convex lower semicontinuous envelope of

$$\mathscr{G}(v) := \begin{cases} G(v), & v = \chi_{\Gamma_u}, \\ \infty, & \text{otherwise}, \end{cases}$$

Here

$$\Gamma_u = \{(x,t) \in \Omega \times \mathbb{R} \mid t < u(x)\}$$

is the lower graph of u, while $v \in L^1(\Omega \times \mathbb{R}; [0,1])$. Then

$$\overline{G}(u) = \overline{\mathscr{G}}(\chi_{\Gamma_u}),$$

and instead of solving (21), one attempts to solve the convex problem

$$\min_{v\in L^1(\Omega\times\mathbb{R};[0,1])} \overline{\mathscr{G}}(v).$$

Observe that v lives in a larger space than u. Although the problem has become convex and more feasible to solve globally than the original one, it has become *bigger*.

Often [24], one can write

$$\overline{\mathscr{G}}(v) = \sup_{\phi\in K} \int_{\Omega\times\mathbb{R}} \nabla^* \phi(x,t) v(x,t)\,d(x,t)$$

for some closed convex set $K \subset C_0(\Omega \times \mathbb{R}; \mathbb{R}^{n+1})$. In some cases, the set K has a numerically realisable analytical expression, although the dimensions of K make the problem even *bigger*.

A particularly important case when K has a simple analytical expression is the for the Mumford-Shah problem (20) [2]. Other problems that can, at least roughly, be handled this way include regularisation by Euler's elastica [18] and multi-label segmentation [108]. Although not exactly fitting this framework, total variation regularisation of discretised manifold-valued data, such as normal fields or direction vectors, can also be performed through convex relaxation in a higher-dimensional space [77]. This approach also covers something as useless, but of utmost mathematical satisfaction, as the smoothing of the path of an ant lost on the Möbius band.

7.2 Decomposition and preconditioning techniques

The idea in domain decomposition is to divide a big problem into small subproblems, solve them, and then combine the solutions. This area is still in its infancy within image processing, although well researched in the context of finite element methods for partial differential equations. The current approaches within the field [57, 58, 48, 49] are still proof-of-concept meta-algorithms that have not replaced the more conventional algorithms discussed in Section 4 and Section 5. They pose difficult (but smaller) problems on each sub-domain. These then have to be solved by one of the conventional algorithms multiple times within the meta-algorithm, which within each of its iterations performs *subspace correction* to glue the solutions together. In case of second-order methods, i.e., if high accuracy is desired, even current domain decomposition techniques may however make problems of previously untractable size tractable.

Depending on the operator A, the first-order methods discussed in Section 4 are however usually easily parallelised within each iteration on multiple CPU cores or on a graphics processing unit (GPU), cf. e.g. [137]. Any advantages of domain decomposition meta-algorithms are therefore doubtful. Intelligent decomposition techniques could however help to reduce the workload within each iteration. This is, roughly, the idea behind stochastic coordinate descent methods, popular in general big data optimisation. We point in particular to [47] for an approach related to FISTA, to [126, 13] for ones related to the ADMM, and to [129, 113, 96, 72, 146, 35, 142] for just a small selection of other approaches. These methods update on each of their iterations only small subsets of unknown variables, even single variables or pixels, and obtain acceleration from local adaptation of step lengths. All of this is done in a random fashion to guarantee fast *expected convergence* on massive data sets. This type of methods form an interesting possibility for image processing.

Stochastic coordinate descent methods generally, however, demand a degree of separability from the problem, limiting the degree of dependence of each variable from other variables. This is necessary both for parallelisation and to prevent lock-

up – to guarantee, statistically, that the randomly chosen variable can be updated without other variables restricting this. This is generally a problem for imaging applications that often lack this level of separability. However, the "coordinate-descent FISTA" of [47], for example, is applicable to the predual formulation (4) of TV denoising. In our preliminary experiments (with Olivier Fercoq and Peter Richtárik), we did not however obtain any acceleration compared to standard FISTA. The problem is that the matrix for the divergence operator in (4) is very uniform. The acceleration features of the current line-up of stochastic gradient descent methods however depend on varying "local curvature" of the problem in terms of local features of the Hessian of the objective. In (4) the Hessian only involves the "uniformly curved" divergence operator, and not the data itself. Therefore, no significant acceleration is obtained, aside from possibly better parallelisation performance, for example on a GPU.

Another alternative to typical domain decomposition techniques is preconditioning – something that has been studied for a long time for general numerical linear algebra, but is still making its inroads into mathematical image processing. Domain decomposition in its per-iteration form can also seen as an approach to preconditioning, of course. Here the idea is to make each iteration cheaper, or, in a sense, to adapt the step sizes spatially. This can be done in the context of the PDHGM, exploiting the proximal point formulation; see [107], where spatial adaptation of the step lengths reportedly significantly improved the performance of the PDHGM. Another recent alternative for which promising performance has been reported, is the use of the conventional Douglas-Rachford splitting method with Gauss-Seidel preconditioning in [19].

Conclusions

In this chapter, we have taken a look into the state-of-the-art of optimisation algorithms suitable for solving mathematical image processing models. Our focus has been on relatively simple first-order splitting methods, as these generally provide the best performance on large-scale images. Moving from FISTA and PDHGM to GIST, we have gradually changed the types of proximal mappings that need to be computed, at the cost of expanding the problem size or reducing theoretical convergence rate. We have also taken a brief look at stochastic gradient descent methods, popular for more general big data problems. At the present stage, such methods are, however, unsuitable for imaging problems. There is thus still significant work to be done in this area— can we come up with an optimisation method that would put mathematically-based state-of-the-art image enhancement models on a pocket camera?

Acknowledgements

The preparation of this chapter was supported by a Prometeo fellowship of the Senescyt (Ecuadorian Ministry of Education, Science, Technology, and Innovation) while the author was at the Centre for Mathematical Modelling (ModeMat), Escuela Politénica, Nacional, Quito, Ecuador.

References

1. Adcock, B., Hansen, A.C., Poon, C., Roman, B.: Breaking the coherence barrier: asymptotic incoherence and asymptotic sparsity in compressed sensing. In: Proc. SampTA 2013 (2013)
2. Alberti, G., Bouchitté, G., Dal Maso, G.: The calibration method for the mumford-shah functional and free-discontinuity problems. Calculus of Variations and Partial Differential Equations **16**(3), 299–333 (2003). DOI 10.1007/s005260100152
3. Ambrosio, L., Fusco, N., Pallara, D.: Functions of Bounded Variation and Free Discontinuity Problems. Oxford University Press (2000)
4. Ambrosio, L., Tortorelli, V.M.: Approximation of functional depending on jumps by elliptic functional via t-convergence. Comm. Pure Appl. Math. **43**(8), 999–1036 (1990). DOI 10.1002/cpa.3160430805
5. Arridge, S.R., Schotland, J.C.: Optical tomography: forward and inverse problems. Inverse Problems **25**(12), 123,010 (2009). DOI 10.1088/0266-5611/25/12/123010
6. Aubert, G., Kornprobst, P.: Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations, 2nd edn. Springer (2006)
7. Bachmayr, M., Burger, M.: Iterative total variation schemes for nonlinear inverse problems. Inverse Problems **25**(10) (2009). DOI 10.1088/0266-5611/25/10/105004
8. Beck, A., Teboulle, M.: Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. IEEE Trans. Image Processing **18**(11), 2419–2434 (2009). DOI 10.1109/TIP.2009.2028250
9. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences **2**(1), 183–202 (2009). DOI 10.1137/080716542
10. Becker, S., Bobin, J., Candès, E.: Nesta: A fast and accurate first-order method for sparse recovery. SIAM Journal on Imaging Sciences **4**(1), 1–39 (2011). DOI 10.1137/090756855
11. Benning, M., Gladden, L., Holland, D., Schönlieb, C.B., Valkonen, T.: Phase reconstruction from velocity-encoded MRI measurements – A survey of sparsity-promoting variational approaches. J. Magnetic Resonance **238**, 26–43 (2014). DOI 10.1016/j.jmr.2013.10.003
12. Bertozzi, A.L., Greer, J.B.: Low-curvature image simplifiers: Global regularity of smooth solutions and Laplacian limiting schemes. Comm. Pure Appl. Math. **57**(6), 764–790 (2004). DOI 10.1002/cpa.20019
13. Bianchi, P., Hachem, W., Iutzeler, F.: A stochastic coordinate descent primal-dual algorithm and applications to large-scale composite optimization. Preprint
14. Blaschke, B., Neubauer, A., Scherzer, O.: On convergence rates for the iteratively regularized Gauss-Newton method. IMA Journal of Numerical Analysis **17**(3), 421–436 (1997)
15. Blum, R., Liu, Z.: Multi-Sensor Image Fusion and Its Applications. Signal Processing and Communications. Taylor & Francis (2005)
16. Bredies, K., Kunisch, K., Pock, T.: Total generalized variation. SIAM J. Imaging Sci. **3**, 492–526 (2011). DOI 10.1137/090769521
17. Bredies, K., Kunisch, K., Valkonen, T.: Properties of L^1-TGV^2: The one-dimensional case. J. Math. Anal Appl. **398**, 438–454 (2013). DOI 10.1016/j.jmaa.2012.08.053

18. Bredies, K., Pock, T., Wirth, B.: A convex, lower semi-continuous approximation of euler's elastica energy. SFB-Report 2013-013, University of Graz (2013)
19. Bredies, K., Sun, H.: Preconditioned Douglas-Rachford splitting methods saddle-point problems with applications to image denoising and deblurring. SFB-Report 2014-002, University of Graz (2014)
20. Bredies, K., Valkonen, T.: Inverse problems with second-order total generalized variation constraints. In: Proc. SampTA 2011 (2011)
21. Buades, A., Coll, B., Morel, J.M.: A non-local algorithm for image denoising. In: IEEE CVPR, pp. 60–65 vol. 2 (2005). DOI 10.1109/CVPR.2005.38
22. Burger, M., Franek, M., Schönlieb, C.B.: Regularized regression and density estimation based on optimal transport. AMRX Appl. Math. Res. Express **2012**(2), 209–253 (2012). DOI 10.1093/amrx/abs007
23. Caselles, V., Chambolle, A., Novaga, M.: The discontinuity set of solutions of the TV denoising problem and some extensions. Multiscale Model. Simul. **6**(3), 879–894 (2008)
24. Chambolle, A.: Convex representation for lower semicontinuous envelopes of functionals in l^1. J. Convex Anal. **8**(1), 149–170 (2001)
25. Chambolle, A., Lions, P.L.: Image recovery via total variation minimization and related problems. Numer. Math. **76**, 167–188 (1997)
26. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vision **40**, 120–145 (2011). DOI 10.1007/s10851-010-0251-1
27. Chan, T., Marquina, A., Mulet, P.: High-order total variation-based image restoration. SIAM J. Sci. Comput. **22**(2), 503–516 (2000). DOI 10.1137/S1064827598344169
28. Chan, T.F., Esedoglu, S.: Aspects of total variation regularized L^1 function approximation. SIAM J. Appl. Math. **65**, 1817–1837 (2005)
29. Chan, T.F., Kang, S.H., Shen, J.: Euler's elastica and curvature-based inpainting. SIAM J. Appl. Math. pp. 564–592 (2002)
30. Chen, K., Lorenz, D.A.: Image sequence interpolation using optimal control. J. Math. Imaging Vision **41**, 222–238 (2011). DOI 10.1007/s10851-011-0274-2
31. Chen, K., Lorenz, D.A.: Image sequence interpolation based on optical flow, segmentation, and optimal control. IEEE Trans. Image Processing **21**(3) (2012). DOI 10.1109/TIP.2011.2179305
32. Chen, X., Nashed, Z., Qi, L.: Smoothing methods and semismooth methods for nondifferentiable operator equations. SIAM J. Numer. Anal. **38**(4), pp. 1200–1216 (2001)
33. Cheney, M., Borden, B.: Problems in synthetic-aperture radar imaging. Inverse Problems **25**(12), 123,005 (2009). DOI 10.1088/0266-5611/25/12/123005
34. Cremers, D., Pock, T., Kolev, K., Chambolle, A.: Convex relaxation techniques for segmentation, stereo and multiview reconstruction. In: Markov Random Fields for Vision and Image Processing. MIT Press (2011)
35. Csiba, D., Qu, Z., Richtárik, P.: Stochastic dual coordinate ascent with adaptive probabilities. Preprint
36. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-d transform-domain collaborative filtering. IEEE Trans. Image Processing **16**(8), 2080–2095 (2007). DOI 10.1109/TIP.2007.901238
37. Dal Maso, G., Fonseca, I., Leoni, G., Morini, M.: A higher order model for image restoration: the one-dimensional case. SIAM J. Math. Anal. **40**(6), 2351–2391 (2009). DOI 10.1137/070697823
38. Daubechies, I., Defrise, M., De Mol, C.: An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Comm. Pure Appl. Math. **57**(11), 1413–1457 (2004). DOI 10.1002/cpa.20042
39. Didas, S., Weickert, J., Burgeth, B.: Properties of higher order nonlinear diffusion filtering. J. Math. Imaging Vision **35**(3), 208–226 (2009). DOI 10.1007/s10851-009-0166-x
40. Douglas Jim, J., Rachford H. H., J.: On the numerical solution of heat conduction problems in two and three space variables. Trans. Amer. Math. Soc. **82**(2), pp. 421–439 (1956)

41. Duval, V., Aujol, J.F., Gousseau, Y.: The TVL1 model: A geometric point of view. Multiscale Model. Simul. **8**, 154–189 (2009)
42. Ehrhardt, M., Arridge, S.: Vector-valued image processing by parallel level sets. IEEE Trans. Image Processing **23**(1), 9–18 (2014). DOI 10.1109/TIP.2013.2277775
43. Engl, H., Hanke, M., Neubauer, A.: Regularization of Inverse Problems. Mathematics and Its Applications. Springer Netherlands (2000)
44. Esser, E., Zhang, X., Chan, T.F.: A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. SIAM J. Imaging Sci. **3**(4), 1015–1046 (2010). DOI 10.1137/09076934X
45. Estrada, F.J., Fleet, D.J., Jepson, A.D.: Stochastic image denoising. In: BMVC, pp. 1–11 (2009). See also http://www.cs.utoronto.ca/ strider/Denoise/Benchmark/ for updated benchmarks
46. Fang, F., Li, F., Zeng, T.: Single image dehazing and denoising: A fast variational approach. SIAM J. Imaging Sci. **7**(2), 969–996 (2014). DOI 10.1137/130919696
47. Fercoq, O., Richtárik, P.: Accelerated, parallel and proximal coordinate descent (2013). Preprint
48. Fornasier, M., Langer, A., Schönlieb, C.B.: A convergent overlapping domain decomposition method for total variation minimization. Numer. Math. **116**(4), 645–685 (2010). DOI 10.1007/s00211-010-0314-7
49. Fornasier, M., Schönlieb, C.: Subspace correction methods for total variation and ℓ_1-minimization. SIAM J. Numer. Anal. **47**(5), 3397–3428 (2009). DOI 10.1137/070710779
50. Gabay, D.: Applications of the method of multipliers to variational inequalities. In: M. Fortin, R. Glowinski (eds.) Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems, Studies in Mathematics and its Applications, vol. 15, pp. 299–331. North-Holland, Amsterdam (1983)
51. Goldstein, T., Osher, S.: The split bregman method for l1-regularized problems. SIAM J. Imaging Sci. **2**(2), 323–343 (2009). DOI 10.1137/080725891
52. Grant, M., Boyd, S.: Graph implementations for nonsmooth convex programs. In: V. Blondel, S. Boyd, H. Kimura (eds.) Recent Advances in Learning and Control, Lecture Notes in Control and Information Sciences, pp. 95–110. Springer-Verlag Limited (2008)
53. Grant, M., Boyd, S.: CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx (2014)
54. Haber, E., Horesh, L., Tenorio, L.: Numerical methods for experimental design of large-scale linear ill-posed inverse problems. Inverse Problems **24**(5), 055,012 (2008). DOI 10.1088/0266-5611/24/5/055012
55. Hanke, M.: A regularizing levenberg-marquardt scheme, with applications to inverse groundwater filtration problems. Inverse Problems **13**(1), 79 (1997). DOI 10.1088/0266-5611/13/1/007
56. He, B., Yuan, X.: Convergence analysis of primal-dual algorithms for a saddle-point problem: From contraction perspective. SIAM J. Imaging Sci. **5**(1), 119–149 (2012). DOI 10.1137/100814494
57. Hintermüller, M., Langer, A.: Non-overlapping domain decomposition methods for dual total variation based image denoising. SFB-Report 2013-014, University of Graz (2013)
58. Hintermüller, M., Langer, A.: Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed l^1/l^2 data-fidelity in image processing. SIAM J. Imaging Sci. **6**(4), 2134–2173 (2013). DOI 10.1137/120894130
59. Hintermüller, M., Stadler, G.: An infeasible primal-dual algorithm for total bounded variation–based inf-convolution-type image restoration. SIAM J. Sci. Comput. **28**(1), 1–23 (2006)
60. Hintermüller, M., Wu, T.: Nonconvex TV^q-models in image restoration: Analysis and a trust-region regularization–based superlinearly convergent solver. SIAM J. Imaging Sci. **6**, 1385–1415 (2013)
61. Hintermüller, M., Wu, T.: A superlinearly convergent R-regularized Newton scheme for variational models with concave sparsity-promoting priors. Comput. Optim. Appl. **57**, 1–25 (2014)

62. Hintermüller, M., Rautenberg, C.N., Hahn, J.: Functional-analytic and numerical issues in splitting methods for total variation-based image reconstruction. Inverse Problems **30**(5), 055,014 (2014). DOI 10.1088/0266-5611/30/5/055014
63. Hintermüller, M., Valkonen, T., Wu, T.: Limiting aspects of non-convex TV^{φ} models (2014). URL http://iki.fi/tuomov/mathematics/tvq.pdf. Submitted
64. Hiriart-Urruty, J.B., Lemaréchal, C.: Convex analysis and minimization algorithms I-II. Springer (1993)
65. Horn, B.K., Schunck, B.G.: Determining optical flow. In: Proc. SPIE, vol. 0281, pp. 319–331 (1981). DOI 10.1117/12.965761
66. Huang, J., Mumford, D.: Statistics of natural images and models. In: IEEE CVPR, vol. 1 (1999)
67. Kaltenbacher, B., Neubauer, A., Scherzer, O.: Iterative Regularization Methods for Nonlinear Ill-Posed Problems. No. 6 in Radon Series on Computational and Applied Mathematics. De Gruyter (2008)
68. Kluckner, S., Pock, T., Bischof, H.: Exploiting redundancy for aerial image fusion using convex optimization. In: M. Goesele, S. Roth, A. Kuijper, B. Schiele, K. Schindler (eds.) Pattern Recognition, *Lecture Notes in Computer Science*, vol. 6376, pp. 303–312. Springer Berlin Heidelberg (2010). DOI 10.1007/978-3-642-15986-2_31
69. Knoll, F., Bredies, K., Pock, T., Stollberger, R.: Second order total generalized variation (TGV) for MRI. Magnetic Resonance in Medicine **65**(2), 480–491 (2011). DOI 10.1002/mrm.22595
70. Knoll, F., Clason, C., Bredies, K., Uecker, M., Stollberger, R.: Parallel imaging with nonlinear reconstruction using variational penalties. Magnetic Resonance in Medicine **67**(1), 34–41 (2012)
71. Kolehmainen, V., Tarvainen, T., Arridge, S.R., Kaipio, J.P.: Marginalization of uninteresting distributed parameters in inverse problems—application to diffuse optical tomography. International Journal for Uncertainty Quantification **1**(1) (2011)
72. Konečný, J., Richtárik, P.: Semi-stochastic gradient descent methods (2013). Preprint
73. Kunisch, K., Hintermüller, M.: Total bounded variation regularization as a bilaterally constrained optimization problem. SIAM J. Imaging Sci. **64**(4), 1311–1333 (2004). DOI 10.1137/S0036139903422784
74. Kunisch, K., Pock, T.: A bilevel optimization approach for parameter learning in variational models. SIAM J. Imaging Sci. **6**(2), 938–983 (2013)
75. Lellmann, J., Lellmann, B., Widmann, F., Schnörr, C.: Discrete and continuous models for partitioning problems. International Journal of Computer Vision **104**(3), 241–269 (2013). DOI 10.1007/s11263-013-0621-4
76. Lellmann, J., Lorenz, D., Schönlieb, C.B., Valkonen, T.: Imaging with Kantorovich-Rubinstein discrepancy. SIAM J. Imaging Sci. **7**, 2833–2859 (2014). DOI 10.1137/140975528. URL http://iki.fi/tuomov/mathematics/krtv.pdf
77. Lellmann, J., Strekalovskiy, E., Koetter, S., Cremers, D.: Total variation regularization for functions with values in a manifold. In: Computer Vision (ICCV), 2013 IEEE International Conference on, pp. 2944–2951 (2013). DOI 10.1109/ICCV.2013.366
78. Lewis, A.: The convex analysis of unitarily invariant matrix functions. J. Convex Anal. **2**(1), 173–183 (1995)
79. Lions, P.L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. SIAM J. Numer. Anal. **16**(6), pp. 964–979 (1979)
80. Lorenz, D.A., Pock, T.: An accelerated forward-backward method for monotone inclusions (2014). Preprint
81. Loris, I.: On the performance of algorithms for the minimization of ℓ_1-penalized functionals. Inverse Problems **25**(3), 035,008 (2009). DOI 10.1088/0266-5611/25/3/035008
82. Loris, I., Verhoeven, C.: On a generalization of the iterative soft-thresholding algorithm for the case of non-separable penalty. Inverse Problems **27**(12), 125,007 (2011). DOI 10.1088/0266-5611/27/12/125007

83. Loris, I., Verhoeven, C.: Iterative algorithms for total variation-like reconstructions in seismic tomography. GEM - International Journal on Geomathematics **3**(2), 179–208 (2012). DOI 10.1007/s13137-012-0036-3
84. de Los Reyes, J.C., Schönlieb, C.B.: Image denoising: Learning noise distribution via PDE-constrained optimization. Inverse Probl. Imaging (2014). To appear
85. de Los Reyes, J.C., Schönlieb, C.B., Valkonen, T.: Optimal parameter learning for higher-order regularisation models (2014). In preparation
86. de Los Reyes, J.C., Schönlieb, C.B., Valkonen, T.: The structure of optimal parameters for image restoration problems (2015). URL http://iki.fi/tuomov/mathematics/interior.pdf. Submitted
87. Lysaker, M., Lundervold, A., Tai, X.C.: Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time. IEEE Trans. Image Processing **12**(12), 1579–1590 (2003). DOI 10.1109/TIP.2003.819229
88. Meyer, Y.: Oscillating patterns in image processing and nonlinear evolution equations. AMS (2001)
89. Morozov, V.A.: On the solution of functional equations by the method of regularization. Soviet Math. Dokldy **7**, 414–417 (1966)
90. Mueller, J.L., Siltanen, S.: Linear and Nonlinear Inverse Problems with Practical Applications. Society for Industrial and Applied Mathematics, Philadelphia, PA (2012). DOI 10.1137/1.9781611972344
91. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Communications on Pure and Applied Mathematics **42**(5), 577–685 (1989). DOI 10.1002/cpa.3160420503
92. Möllenhoff, T., Strekalovskiy, E., Möller, M., Cremers, D.: The primal-dual hybrid gradient method for semiconvex splittings. arXiv preprint arXiv:1407.1723 (2014)
93. Möller, M., Burger, M., Dieterich, P., Schwab, A.: A framework for automated cell tracking in phase contrast microscopic videos based on normal velocities. Journal of Visual Communication and Image Representation **25**(2), 396–409 (2014). DOI 10.1016/j.jvcir.2013.12.002. URL http://www.sciencedirect.com/science/article/pii/S1047320313002162
94. Nesterov, Y.: A method of solving a convex programming problem with convergence rate $O(1/k^2)$. Soviet Mathematics Doklady **27**(2), 372–376 (1983)
95. Nesterov, Y.: Smooth minimization of non-smooth functions. Math. Program. **103**(1), 127–152 (2005). DOI 10.1007/s10107-004-0552-5
96. Nesterov, Y.: Efficiency of coordinate descent methods on huge-scale optimization problems. SIAM J. Optim. **22**(2), 341–362 (2012). DOI 10.1137/100802001
97. Nocedal, J., Wright, S.: Numerical Optimization. Springer Series in Operations Research and Financial Engineering. Springer (2006)
98. Ochs, P., Chen, Y., Brox, T., Pock, T.: iPiano: Inertial proximal algorithm for non-convex optimization. arXiv preprint arXiv:1404.4805 (2014)
99. Ochs, P., Dosovitskiy, A., Brox, T., Pock, T.: An iterated l1 algorithm for non-smooth non-convex optimization in computer vision. In: IEEE CVPR (2013)
100. Osher, S., Burger, M., Goldfarb, D., Xu, J., Yin, W.: An iterative regularization method for total variation-based image restoration. Multiscale Model. Simul. **4**(2), 460–489 (2005). DOI 10.1137/040605412
101. Pan, X., Sidky, E.Y., Vannier, M.: Why do commercial ct scanners still employ traditional, filtered back-projection for image reconstruction? Inverse Problems **25**(12), 123,009 (2009). DOI 10.1088/0266-5611/25/12/123009
102. Papafitsoros, K., Bredies, K.: A study of the one dimensional total generalised variation regularisation problem (2013). Preprint
103. Papafitsoros, K., Schönlieb, C.B.: A combined first and second order variational approach for image reconstruction. J. Math. Imaging Vision **48**(2), 308–338 (2014). DOI 10.1007/s10851-013-0445-4
104. Papafitsoros, K., Valkonen, T.: Asymptotic behaviour of total generalised variation. In: Fifth International Conference on Scale Space and Variational Methods in Computer Vision (SSVM) (2015). URL http://iki.fi/tuomov/mathematics/ssvm2015-40.pdf. Accepted

105. Passty, G.B.: Ergodic convergence to a zero of the sum of monotone operators in hilbert space. J. Math. Anal Appl. **72**(2), 383–390 (1979). DOI 10.1016/0022-247X(79)90234-8
106. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE TPAMI **12**(7), 629–639 (1990). DOI 10.1109/34.56205
107. Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In: Computer Vision (ICCV), 2011 IEEE International Conference on, pp. 1762 –1769 (2011). DOI 10.1109/ICCV.2011.6126441
108. Pock, T., Chambolle, A., Cremers, D., Bischof, H.: A convex relaxation approach for computing minimal partitions. In: IEEE CVPR, pp. 810–817 (2009). DOI 10.1109/CVPR.2009.5206604
109. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An algorithm for minimizing the mumford-shah functional. In: Computer Vision, 2009 IEEE 12th International Conference on, pp. 1133–1140 (2009). DOI 10.1109/ICCV.2009.5459348
110. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: Global solutions of variational models with convex regularization. SIAM Journal on Imaging Sciences **3**(4), 1122–1145 (2010). DOI 10.1137/090757617
111. Qi, L., Sun, J.: A nonsmooth version of newton's method. Math. Program. **58**(1-3), 353–367 (1993). DOI 10.1007/BF01581275
112. Ranftl, R., Pock, T., Bischof, H.: Minimizing tgv-based variational models with non-convex data terms. In: A. Kuijper, K. Bredies, T. Pock, H. Bischof (eds.) Scale Space and Variational Methods in Computer Vision, *Lecture Notes in Computer Science*, vol. 7893, pp. 282–293. Springer Berlin Heidelberg (2013). DOI 10.1007/978-3-642-38267-3_24
113. Richtárik, P., Takáč, M.: Parallel coordinate descent methods for big data optimization. Mathematical Programming pp. 1–52 (2015). DOI 10.1007/s10107-015-0901-6
114. Ring, W.: Structural properties of solutions to total variation regularization problems. ESAIM: Math. Model. Numer. Anal. **34**, 799–810 (2000). DOI 10.1051/m2an:2000104
115. Rockafellar, R.T.: Convex Analysis. Princeton University Press (1972)
116. Rockafellar, R.T.: Monotone operators and the proximal point algorithm. SIAM J. Optim. **14**(5), 877–898 (1976). DOI 10.1137/0314056
117. Rockafellar, R.T., Wets, R.J.B.: Variational Analysis. Springer (1998)
118. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60**, 259–268 (1992)
119. Sawatzky, A., Brune, C., Müller, J., Burger, M.: Total variation processing of images with poisson statistics. In: X. Jiang, N. Petkov (eds.) Computer Analysis of Images and Patterns, *Lecture Notes in Computer Science*, vol. 5702, pp. 533–540. Springer Berlin Heidelberg (2009). DOI 10.1007/978-3-642-03767-2_65
120. Schuster, T., Kaltenbacher, B., Hofmann, B., Kazimierski, K.: Regularization Methods in Banach Spaces. Radon Series on Computational and Applied Mathematics. De Gruyter (2012)
121. Setzer, S.: Operator splittings, bregman methods and frame shrinkage in image processing. Int. J. Comput. Vis. **92**(3), 265–280 (2011). DOI 10.1007/s11263-010-0357-3
122. Shen, J., Kang, S., Chan, T.: Euler's elastica and curvature-based inpainting. SIAM J. Appl. Math. **63**(2), 564–592 (2003). DOI 10.1137/S0036139901390088
123. Stathaki, T.: Image Fusion: Algorithms and Applications. Elsevier Science (2011)
124. Strang, G., Nguyen, T.: Wavelets and filter banks. Wellesley Cambridge Press (1996)
125. Sun, D., Han, J.: Newton and Quasi-Newton methods for a class of nonsmooth equations and related problems. SIAM J. Optim. **7**(2), 463–480 (1997). DOI 10.1137/S1052623494274970
126. Suzuki, T.: Stochastic dual coordinate ascent with alternating direction multiplier method (2013). Preprint
127. Tournier, J.D., Mori, S., Leemans, A.: Diffusion tensor imaging and beyond. Magnetic Resonance in Medicine **65**(6), 1532–1556 (2011). DOI 10.1002/mrm.22924
128. Tseng, P.: Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. SIAM J. Control Optim. **29**(1), 119–138 (1991). DOI 10.1137/0329006

129. Tseng, P., Yun, S.: A coordinate gradient descent method for nonsmooth separable minimization. Math. Program. **117**(1-2), 387–423 (2009). DOI 10.1007/s10107-007-0170-0
130. Valkonen, T.: Transport equation and image interpolation with SBD velocity fields. J. Math. Pures Appl. **95**, 459–494 (2011). DOI 10.1016/j.matpur.2010.10.010
131. Valkonen, T.: The jump set under geometric regularisation. Part 2: Higher-order approaches (2014). Submitted
132. Valkonen, T.: A method for weighted projections to the positive definite cone. Optimization (2014). DOI 10.1080/02331934.2014.929680. Published online 24 Jun 2014
133. Valkonen, T.: A primal-dual hybrid gradient method for non-linear operators with applications to MRI. Inverse Problems **30**(5), 055,012 (2014). DOI 10.1088/0266-5611/30/5/055012
134. Valkonen, T.: The jump set under geometric regularisation. Part 1: Basic technique and first-order denoising. SIAM J. Math. Anal. **47**(4), 2587–2629 (2015). DOI 10.1137/140976248. URL http://iki.fi/tuomov/mathematics/jumpset.pdf
135. Valkonen, T., Bredies, K., Knoll, F.: Total generalised variation in diffusion tensor imaging. SIAM J. Imaging Sci. **6**(1), 487–525 (2013). DOI 10.1137/120867172
136. Valkonen, T., Knoll, F., Bredies, K.: TGV for diffusion tensors: A comparison of fidelity functions. J. Inverse Ill-Posed Probl. **21**, 355–377 (2013). DOI 10.1515/jip-2013-0005. Special issue for IP:M&S 2012, Antalya, Turkey
137. Valkonen, T., Liebmann, M.: GPU-accelerated regularisation of large diffusion tensor volumes. Computing **95**, 771–784 (2013). DOI 10.1007/s00607-012-0277-x. Special issue for ESCO 2012, Pilsen, Czech Republic
138. Vese, L.A., Chan, T.F.: A multiphase level set framework for image segmentation using the Mumford and Shah model. Int. J. Comput. Vis. **50**(3), 271–293 (2002). DOI 10.1023/A:1020874308076
139. Vese, L.A., Osher, S.J.: Modeling textures with total variation minimization and oscillating patterns in image processing. J. Sci. Comput. **19**(1-3), 553–572 (2003)
140. Vogel, C., Oman, M.: Iterative methods for total variation denoising. SIAM J. Sci. Comput. **17**(1), 227–238 (1996). DOI 10.1137/0917016
141. Wernick, M., Aarsvold, J.: Emission Tomography: The Fundamentals of PET and SPECT. Elsevier Science (2004)
142. Wright, S.: Coordinate descent algorithms. Math. Program. **151**(1), 3–34 (2015). DOI 10.1007/s10107-015-0892-3
143. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. ACM Computing Surveys (CSUR) **38**(4), 13 (2006)
144. Yin, W., Osher, S., Goldfarb, D., Darbon, J.: Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing. SIAM J. Imaging Sci. **1**(1), 143–168 (2008). DOI 10.1137/070703983
145. Zhao, F., Noll, D., Nielsen, J.F., Fessler, J.: Separate magnitude and phase regularization via compressed sensing. IEEE Trans. Medical Imaging **31**(9), 1713–1723 (2012). DOI 10.1109/TMI.2012.2196707
146. Zhao, P., Zhang, T.: Stochastic optimization with importance sampling (2014). Preprint
147. Zhu, M., Chan, T.: An efficient primal-dual hybrid gradient algorithm for total variation image restoration. UCLA CAM Report (2008)