Tomographic image reconstruction using training images

Soltani, Sara; Andersen, Martin Skovgaard; Hansen, Per Christian

Published in:
Journal of Computational and Applied Mathematics

Link to article, DOI:
10.1016/j.cam.2016.09.019

Publication date:
2017

Document Version
Peer reviewed version

Tomographic Image Reconstruction using Training Images[☆]

Sara Soltani[a], Martin S. Andersen[b], Per Christian Hansen[b]

[a] *Fingerprint Cards AB, Lyskær 3CD, DK-2730 Herlev, Denmark*
[b] *Department of Applied Mathematics and Computer Science, Technical University of Denmark, DK-2800 Kgs. Lyngby, Denmark.*

Abstract

We describe and examine an algorithm for tomographic image reconstruction where prior knowledge about the solution is available in the form of training images. We first construct a non-negative dictionary based on prototype elements from the training images; this problem is formulated within the framework of sparse learning as a regularized non-negative matrix factorization. Incorporating the dictionary as a prior in a convex reconstruction problem, we then find an approximate solution with a sparse representation in the dictionary. The dictionary is applied to non-overlapping patches of the image, which reduces the computational complexity compared to previous formulations. Computational experiments clarify the choice and interplay of the model parameters and the regularization parameters, and we show that in few-projection low-dose settings our algorithm is competitive with total variation regularization and tends to include more texture and more correct edges.

Keywords: Tomography, Dictionary learning, Inverse problems, Regularization, Sparse representation, Image reconstruction.
2010 MSC: Primary: 65F22, Secondary: 65K10.

1. Introduction

Computed tomography (CT) is a technique to compute an image of the interior of an object from measurements obtained by sending X-rays through the object and recording the damping of each ray. CT is used routinely in medical imaging, materials science, nondestructive testing and many other applications.

CT is an inverse problem [30] and it is challenging to obtain sharp and reliable reconstructions in low-dose measurements where we face underdetermined systems of equations, because we must limit the accumulated amount of X-rays for health reasons or because measurement time is limited. In these circumstances the classic methods of CT, such as filtered back projection [20] and algebraic reconstruction techniques [14], are often incapable of producing satisfactory reconstructions because they fail to incorporate adequate prior information [3]. To overcome these difficulties it is necessary to incorporate prior information about the solution that can compensate for the lack of data.

A popular prior is that the image is piecewise constant, leading to total variation (TV) regularization schemes [21], [38]. These methods can be very powerful when the solution is approximately composed of homogeneous regions separated by sharp boundaries.

A completely different approach is to use prior information in the form of "training images" that characterize the geometrical or visual features of interest, e.g., from high-accuracy reconstructions or from pictures of specimen slices. The goal of this work is to elaborate on this approach. In particular we consider the two-stage framework where the most important features of the training data are first extracted and then integrated in the reconstruction problem.

A natural way to extract and represent prior information from training images is to form a *dictionary* that sparsely encodes the information [31]. Learning the dictionary from given training data appears to be very suited for incorporating priors that are otherwise difficult to formulate in a closed form, such as image texture. Dictionary learning — combined with sparse representation [5, 9, 37] — is now used in many image processing areas including denoising

[☆] This work is part of the project HD-Tomo funded by Advanced Grant No. 291405 from the European Research Council.
Email addresses: sara.soltani@fingerprints.com (Sara Soltani), mskan@dtu.dk (Martin S. Andersen), pcha@dtu.dk (Per Christian Hansen)

[7], [24], inpainting [27], and deblurring [25]. Elad and Ahron [10] address the image denoising problem using a process that combines dictionary learning and reconstruction. They use a dictionary trained from a noise-free image using the K-SVD algorithm [1] combined with an adaptive dictionary trained on patches of the noisy image.

The use of dictionary learning in tomographic imaging has also emerged recently, e.g., in X-ray CT [12, 39, 41], magnetic resonance imaging [16, 32], electron tomography [26], positron emission tomography [8], and phase-contrast tomography [29]. Two different approaches have emerged — either one constructs the dictionary from the given data in a joint learning-reconstruction algorithm [8, 16, 26, 32], or one constructs the dictionary from training images in a separate step before the reconstruction [12, 29, 39, 41]. Most of these works use K-SVD to learn the dictionary (except [12] that uses an "online dictionary learning method" [28]), and all the methods regularize the reconstruction by means of a penalty that is applied to a patch around every pixel in the image. In other words, all patches in the reconstruction are required be close to the subspace spanned by the dictionary images. While all these methods perform better than classical reconstruction methods, they show no significant improvement over the TV-regularized approach.

In simultaneous learning and reconstruction, where the dictionary is learned from the given data, the prior is purely data-driven. Hence, one can argue that it violates a fundamental principle of inverse problems where a data-independent prior is incorporated to eliminate unreasonable models that fit the data. For this reason we prefer to separate the two steps (which requires that reliable training images are available). We describe and examine a two-stage framework where we first construct a dictionary that contains prototype elements from these images, and then we use the dictionary as a prior to regularize the reconstruction problem via computing a solution that has a sparse representation in the dictionary.

Our two-stage algorithm is inspired by the work in [12] and, to some extent, [39]. The algorithm in [12] is tested on a simple and ideal tomography setup with no noise in the data, and in [39] the dictionary is trained from an image reconstructed by a high-dose X-ray exposure and then used to reconstruct the same image with fewer X-ray projections.

The focus of this paper is a new formulation of tomographic reconstruction where training images are used as a strong prior. Our algorithm utilizes the dictionary in a different way than previous formulations, by using *non-overlapping* blocks of the image which reduces the number of unknowns and thus the computational work. We use state-of-the-art numerical optimization methods to handle the large-scale optimization problems, and we apply this algorithm applied to underdetermined problems where a strong prior is necessary. We perform comprehensive studies of the influence of the learned dictionary structure and the dictionary parameters on the CT reconstruction, and we compare our algorithm with both classical methods and with TV regularization.

Our paper is organized as follows. In section 2 we briefly discuss dictionary learning methods and present a framework for solving the image reconstruction problem using dictionaries, and in Section 3 we describe the implementation details of algorithm. Section 4 presents careful numerical experiments where we study the influence of the algorithm and design parameters. Section 5 summarizes our work. We use the following notation, where A is an arbitrary matrix:

$$\|A\|_{\mathrm{F}} = \left(\sum_{ij} A_{ij}^2\right)^{1/2}, \quad \|A\|_{\mathrm{sum}} = \sum_{ij} |A_{ij}|, \quad \|A\|_{\max} = \max_{ij} |A_{ij}|.$$

2. The Reconstruction Framework

X-ray CT is based on the principle that if we send X-rays through an object and measure the damping of each ray then, with infinitely many rays, we can perfectly reconstruct the object. The attenuation of an X-ray is proportional to the object's attenuation coefficient, as described by Lambert-Beer's law [6, §2.3.1]. We divide the domain onto pixels whose unknown non-negative attenuation coefficients are organized in the vector $x \in \mathbb{R}^n$. Similarly we organize the measured damping of the rays into the vector $b \in \mathbb{R}^m$. Then we obtain a linear system of equations $A\,x = b$ with a large sparse *system matrix* A governed solely by the geometry of the measurements: element a_{ij} is the length of the ith ray passing through pixel j, and the matrix is sparse because each ray only hits a small number of pixels [30].

The matrix A is ill-conditioned, and often rank deficient, due to the ill-posedness of the underlying inverse problem and therefore the solution is very sensitive to noise in the data b. For this reason, a simple least squares approach with non-negativity constraints fails to produce a meaningful solution, and we must use regularization to incorporate prior information about the solution [13].

This work is concerned with underdetermined problems where $m < n$, and the need for regularization is even more pronounced. Classical reconstruction methods such as filtered back projection and algebraic iterative methods are not suited for these problems because they fail to incorporate enough prior information. TV regularization, which is suited for edge-preserving reconstructions, takes the form

$$\min_x \quad \tfrac{1}{2}\,\|A\,x - b\|_2^2 + \lambda_{\mathrm{TV}} \sum_{1\le i\le n} \left\|D_i^{\mathrm{fd}} x\right\|_2 \qquad \text{subject to} \qquad x \ge 0, \tag{1}$$

where we have included a non-negativity constraint; $D_i^{\mathrm{fd}} x$ is a finite-difference approximation of the gradient at pixel i, and $\lambda_{\mathrm{TV}} > 0$ is a regularization parameter. TV methods produce images whose pixel values are clustered into regions with almost constant intensity [35], with the result that textural images tend to be over-smoothed (except for the sharp edges). Another drawback is that the TV problem (1) tends to produce reconstructions whose intensities are incorrect [35].

Our goal is to incorporate prior information — e.g., about texture — from a set of training images. We focus on formulating and finding a learned dictionary W from the training images and solving the tomography problem such that $x = W\alpha$ is a sparse linear combination of the dictionary elements (the columns of W). We build on ideas from sparse approximation [5, 9, 37] which seeks an approximate representation of a signal/image using a linear combination of a few known basis elements.

As mentioned in the Introduction, some works use a joint formulation that combines the dictionary learning problem and the reconstruction problem into one optimization problem, i.e., the dictionary is learned from the given noisy data. This corresponds to a "bootstrap" situation where one creates the prior as part of the solution process. Our work is different: we use a prior that is already available in the form of a set of training images, and we use this prior to regularize the reconstruction problem. To do this, we use a two-stage algorithm where we first compute the dictionary from the given training images, and then we use the dictionary to compute the reconstruction.

The dictionary W should comprise all the important features of the desired solution. A learned dictionary — while computationally more expensive than a fixed dictionary — has the advantage that it is tailored to the characteristics of the desired solution and optimized for the training images. Dictionary learning is a way to summarize and represent a large number of training images into fewer elements and, at the same time, compensate for noise or other errors in these images. The learned dictionary should be robust to irrelevant features, and the number of training images should be large enough to ensure that all image features are represented; hence dictionaries are typically overcomplete.

Using training images of the same size as the image to be reconstructed would require a huge number of training images and lead to an enormous dictionary. All algorithms therefore use a *patch dictionary* D learned from patches of the training images. But contrary to previous algorithms that apply a dictionary-based regularization based on overlapping patches around every pixel in the image, we divide the reconstruction into non-overlapping blocks of the same size as the patches and use the dictionary D within each block (ensuring that we limit blocking effects); conceptually this corresponds to building a *global dictionary* W from the patch dictionary D.

Let the patches be of size $P \times Q$, and let the matrix $Y \in \mathbb{R}_+^{p\times t}$ consist of t training image patches arranged as vectors of length $p = PQ$. Then the dictionary learning problem can be viewed as the problem of approximating the training matrix as a product of two matrices, $Y \approx DH$, where $D \in \mathbb{R}^{p\times s}$ is the dictionary of s dictionary image patches (the columns of D), and $H \in \mathbb{R}^{s\times t}$ contains information about the approximation of each of the training image patches. Such a decomposition is clearly not unique, so we must incorporate further requirements to "shape" the patch dictionary D and the representation matrix H.

Imposing norm and/or non-negativity constraints on the elements of D and H or imposing sparsity constraint on matrix H are widely used in unsupervised learning. We take the same approach, and thus our generic dictionary learning problem takes the form:

$$\min_{D,H} \quad \mathcal{L}_{\mathrm{dic}}(Y, DH) + \Phi_{\mathrm{dic}}(D) + \Phi_{\mathrm{rep}}(H). \tag{2}$$

Here, the misfit of the factorization approximation is measured by the loss function $\mathcal{L}_{\mathrm{dic}}$, while the priors on the patch dictionary D and the representation matrix H are taken into account by the regularization functions Φ_{dic} and Φ_{rep}.

The dictionary learning problem (2) is a non-convex optimization problem. If we choose the functions $\mathcal{L}_{\mathrm{dic}}$, Φ_{dic} and Φ_{rep} to be convex, then the optimization problem in (2) is not jointly convex in (D, H), but it is convex with

respect to each variable D or H when the other is fixed. A natural way to find a local minimum is therefore to use an alternating approach, first minimizing over H with D fixed, and then minimizing over D with H fixed.

Various dictionary learning methods proposed in the literature share the same overall structure but they consider different priors when formulating the dictionary learning problem. Examples of such methods include, but are not limited to, non-negative matrix factorization [22], the method of optimal directions [11], K-means clustering [18] and its generalization K-SVD [10], and the online dictionary learning method [28]. The methods in [19] and [23] are designed for training data corrupted by additive noise; but this it is not important for our work.

Having computed the patch dictionary D and formed the corresponding global dictionary W, the second step is to solve the reconstruction problem. Using ideas from sparse approximation, we compute a solution $x = W\alpha$ where α solves the problem

$$\min_{\alpha} \quad \mathcal{L}_{\text{rec}}(AW\alpha, b) + \Phi_{\text{SP}}(\alpha) + \Phi_{\text{IP}}(W\alpha), \tag{3}$$

in which the data fidelity is measured by the loss function $\mathcal{L}_{\text{rec}}$ and regularization is imposed via penalty functions. Specifically, the function Φ_{SP} enforces the Sparsity Prior on α, often formulated in terms of a sparsity inducing norm, while the function Φ_{IP} enforces the Image Prior. If we choose the three functions $\mathcal{L}_{\text{rec}}$, Φ_{SP} and Φ_{IP} to be convex, then the problem formulation (3) can be solved by means of convex optimization methods. Given a solution $\alpha^\star$ to (3) we compute the solution as $x^\star = W\alpha^\star$. In Section 4 we illustrate with numerical examples that the sparsity penalties in (2) and (3) tend to have a regularizing effect on the reconstruction.

3. Details of Formulation and Implementation

Recall that the proposed framework for dictionary-based tomographic reconstruction consists of two conceptual steps: (i) computing a dictionary (using techniques from machine learning), and (ii) computing a reconstruction composed of images from the dictionary. In this section we describe one of many ways to efficiently implement such a scheme. We pose the dictionary-learning problem as a so-called non-negative sparse coding problem, and we use least squares optimization with non-negative variables and 1-norm regularization to compute a reconstruction.

3.1. The Dictionary Learning Problem

Dictionary learning problems of the form (2) are generally non-convex optimization problems due to the bilinear term DH where both D and H are unknown. Applying a convergent iterative optimization method therefore does not guarantee that we find a global minimum (only a local stationary point). To obtain a good dictionary, we must be careful when choosing the loss function $\mathcal{L}_{\text{dic}}$ and the penalties Φ_{dic} and Φ_{rep} on D and H, and we must also pay attention to implementation issues such as the starting point; see the Appendix for details.

A non-negative matrix factorization (NMF) has the ability to extract meaningful factors [22], and with non-negative elements in D its columns represent a basis of images. Similarly, having non-negative elements in H corresponds to each training image being represented as a conic combination of dictionary images, and the representation itself is therefore non-negative. NMF often works well in combination with sparsity heuristics [15] which in our application translates to training image patches being represented as a conic combination of a small number of dictionary elements (basis images).

The dictionary learning problem that we will use henceforth takes the form of non-negative sparse coding [15] of a non-negative data matrix Y:

$$\min_{D,H} \quad \tfrac{1}{2}\,\|Y - DH\|_{\text{F}}^2 + \lambda\,\|H\|_{\text{sum}} \qquad \text{s.t.} \qquad D \in \mathcal{D},\ H \in \mathbb{R}_+^{s\times t}, \tag{4}$$

where the set $\mathcal{D}$ is compact and convex and $\lambda \geq 0$ is a regularization parameter that controls the sparsity-inducing penalty $\|H\|_{\text{sum}}$. This problem is an instance of the more general formulation (2) if we define

$$\mathcal{L}_{\text{dic}}(Y, DH) = \tfrac{1}{2}\|Y - DH\|_{\text{F}}^2$$

and

$$\Phi_{\text{dic}}(D) = I_{\mathcal{D}}(D), \qquad \Phi_{\text{rep}}(H) = I_{\mathbb{R}_+^{s\times t}}(H) + \lambda\|H\|_{\text{sum}}\,,$$

where $I_{\mathcal{D}}$ denotes the indicator function of the set $\mathcal{D}$. Note that the loss function $\mathcal{L}_{\text{dic}}$ is invariant under a scaling $D \mapsto \zeta D$ and $H \mapsto \zeta^{-1} H$ for $\zeta > 0$, and letting $\zeta \to \infty$ implies that $\Phi_{\text{rep}}(\zeta^{-1} H) \to 0$ and $\|\zeta D\| \to \infty$ if D is nonzero. This means that $\mathcal{D}$ must be compact to ensure that the problem has well-defined minima. Here we will consider two different definitions of the set $\mathcal{D}$, namely

$$\mathcal{D}_\infty \equiv \{D \in \mathbb{R}_+^{p\times s} \,|\, \|d_j\|_\infty \le 1\} \quad \text{and} \quad \mathcal{D}_2 \equiv \{D \in \mathbb{R}_+^{p\times s} \,|\, \|d_j\|_2 \le \sqrt{p}\}.$$

The set $\mathcal{D}_\infty$ corresponds to box constraints, and $\mathcal{D}_2$ is a spherical sector of the 2-norm ball with radius $\sqrt{p}$. As we will see in the next section, the use of $\mathcal{D}_\infty$ as a prior gives rise to binary-looking images (corresponding to the vertices of $\mathcal{D}_\infty$) whereas $\mathcal{D}_2$ gives rise to more "natural looking" images.

We emphasize an important difference between the classical K-SVD method and our method. While K-SVD requires that we explicitly set the sparsity level, in our approach we affect sparsity implicitly through 1-norm regularization and via the regularization parameter λ.

We use the Alternating Direction Method of Multipliers (ADMM) [4] to compute an approximate local minimizer of (4). Learning the dictionary with the ADMM method has the advantages that the updates are cheap to compute, making the method suited for large-scale problems. The implementation details are described in the Appendix.

3.2. The Reconstruction Problem

Recall that we formulate the CT problem as $Ax \approx b$, where b contains the noisy data and A is the system matrix. The vector x represents an $M \times N$ image of absorption coefficients, and these coefficients must be nonnegative to have physical meaning. Hence we must impose a nonnegativity constraint on the solution.

Let us turn to the reconstruction problem based on the patch dictionary D and the formulation (3). For ease of our presentation we assume that the image size $M \times N$ is a multiple of the patch size $P \times Q$, and we partition the image into an $(M/P) \times (N/Q)$ array of non-overlapping blocks or patches represented by the vectors $x_j \in \mathbb{R}^p$ for $j = 1, \ldots, q = (M/P)(N/Q)$. The advantage of using non-overlapping blocks, compared to overlapping blocks, is that we avoid over-smoothing the image textures when averaging over the overlapping regions, and it requires less computing time.

Each block of x is expressed as a conic combination of dictionary images, and hence the dictionary prior is expressed as

$$x = \Pi\, W\alpha, \quad W = (I \otimes D), \qquad \alpha \ge 0, \tag{5}$$

where Π is a permutation matrix which ensures that we reconstruct the image block by block, W is the global dictionary, and

$$\alpha = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_q \end{pmatrix} \in \underbrace{\mathbb{R}^s \times \cdots \times \mathbb{R}^s}_{q \text{ times}}$$

is a vector of coefficients for each of a total of q blocks. With this non-overlapping formulation, it is straightforward to determine the numebr of unknowns in the problem (8). The length of α is $sq = n\, s/p$ which is equal to the product of the over-representation factor s/p and the number of pixels n in the image.

In pursuit of a nonnegative image x, we impose the constraint that the vector α should be nonnegative. This implies that each block x_j of x lies inside a polyhedral cone

$$C = \{Dz \,|\, z \in \mathbb{R}_+^s\} \subseteq \mathbb{R}_+^p \tag{6}$$

as illustrated in Figure 1. Clearly, if the dictionary contains the standard basis of $\mathbb{R}^p$ then C is equivalent to the entire nonnegative orthant in $\mathbb{R}^p$. However, if the cone C is a proper subset of $\mathbb{R}_+^p$, then not all nonnegative images have an exact representation in C, and hence the constraints $x_j \in C$ may have a regularizing effect even without a sparsity prior on α. This can also be motivated by the fact that the faces of the cone C consist of images x_j that can be represented as a conic combination of at most $p - 1$ dictionary images.

Adding a sparsity prior on α, in addition to nonnegativity constraints, corresponds to the assumption that x_j can be expressed as a conic combination of a small number of dictionary images and hence provides additional regularization. We include a 1-norm regularizer in our reconstruction problem as the standard approximate sparsity prior on α.

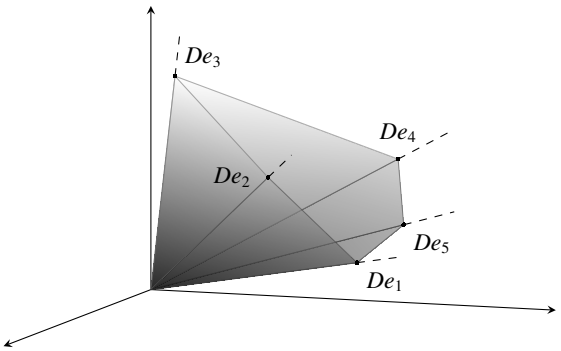


Figure 1: Polyhedral cone in $\mathbb{R}^p_+$ spanned by five nonnegative dictionary elements, where e_i denotes the ith canonical unit vector in $\mathbb{R}^s$.

Reconstruction based on non-overlapping blocks often gives rise to block artifacts in the reconstruction because the objective in the reconstruction problem does not penalize jumps across the boundaries of neighboring blocks. To mitigate this type of artifact, we add a penalty term that discourages such jumps. We choose a penalty of the form

$$\psi(z) = \tfrac{1}{2}\,\|Lz\|_2^2/\ell, \quad \ell = M(M/P-1) + N(N/Q-1) \tag{7}$$

where L is a matrix such that Lz is a vector with finite-difference approximations of the directional derivatives across the block boundaries. The factor ℓ is the total number of pixels along the boundaries of the blocks in the image.

The constrained least squares reconstruction problem is then given by

$$\begin{array}{ll} \text{minimize}_\alpha & \frac{1}{2}\frac{1}{m}\|A\Pi(I\otimes D)\alpha - b\|_2^2 + \mu\,\frac{1}{q}\|\alpha\|_1 + \delta^2\,\psi(\Pi(I\otimes D)\alpha) \\ \text{subject to} & \alpha \geq 0 \end{array} \tag{8}$$

with regularization parameters $\mu, \delta > 0$. We normalize the problem formulation by i) division of the squared residual norm by the number of measurement m, ii) division of the 1-norm of α by the number of blocks q, and iii) division by ℓ in the function ψ. Problem (8) is convex and it is an instance of a sparse approximation problem similar to formulations studied in [10].

4. Numerical Experiments

In this section we use numerical examples to demonstrate and quantify the behavior of our two-stage algorithm and evaluate the computed reconstructions. In particular we explore the influence of the dictionary structure and its parameters (number of elements, patch sizes) on the reconstruction, in order to illustrate the role of the learned dictionary.

The underlying idea is to compute a regularized least squares fit in which the solution is expressed in terms of the dictionary, and hence it lies in the cone C (6) defined by the dictionary elements. Hence there are two types of errors in the reconstruction process. Typically, the exact image does not lie in the cone C, leading to an *approximation error*. Moreover, we encounter a *regularization error* due to the combination of the error present in the data and the regularization scheme.

In the learning stage we use a set of images which are similar to the ones we wish to reconstruct. The ground-truth or exact image x^{exact} is not contained in the training set, so that we avoid committing an inverse crime. All images are gray-level and scaled in the interval $[0, 1]$.

All experiments were run in MATLAB (R2011b) on a 64-bit Linux system. The reconstruction problems are solved using the software package TFOCS (Templates for First-Order Conic Solvers) [2]. We compare with TV reconstructions computed by means of the MATLAB software TVReg [17], with filtered back projection solutions computed by means of MATLAB's "iradon" function, and solutions computed by means of the algebraic reconstruction technique (ART, also known as Kaczmarz's method) with nonnegativety constraints implemented in the MATLAB package AIR Tools [14]. (We did not compare with Krylov subspace methods because they are inferior to ART for images with sharp edges.)

4.1. The Test Image and the Tomographic Test Problem

The test images used in Sections 4.2–4.5 are square patches from a high-resolution photo of peppers with uneven surfaces resembling texture, making them interesting test images for studies of the reconstruction of textures and structures with sharp boundaries. Figure 2 shows the 1600×1200 high-resolution image and the exact image of dimensions $M \times N = 200 \times 200$. This size allows us to perform many numerical experiments in a reasonable amount of time; we demonstrate the performance of our algorithm on a larger test problem in Section 4.6.

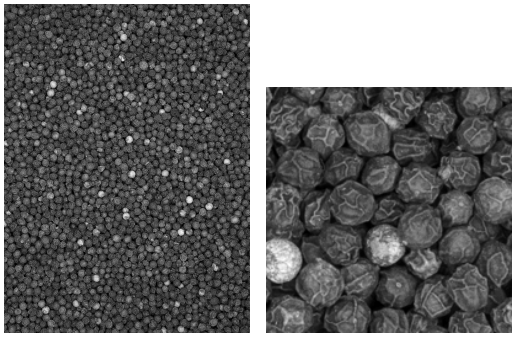


Figure 2: Left: the high-resolution image from which we obtain the training image patches. Right: the 200×200 exact image x^{exact}.

All test problems represent a parallel-beam tomographic measurement, and we use the function `paralleltomo` from AIR Tools [14] to compute the system matrix A. The data associated with a set of parallel rays is called a projection, and number of rays in each projection is given by $N_{\text{r}} = \lfloor \sqrt{2}N \rfloor$. If the total number of projections is N_{p} then the number of rows in A is $m = N_{\text{r}}N_{\text{p}}$ while the number of columns is $n = MN$. Recall that we are interested in scenarios with a small number of projections. The exact data is generated with the forward model after which we add Gaussian white noise, i.e., $b = Ax^{\text{exact}} + e$.

4.2. Studies of the Dictionary Learning Stage

A good dictionary should preserve the structural information of the training images as much as possible and, at the same time, admit a sparse representation as well as a small representation error. These requirements are related to the number of dictionary elements, i.e., the number of columns s in the matrix $D \in \mathbb{R}^{p \times s}$. Since we want a compressed representation of the training images we choose s such that $p \leq s \ll t$, and the precise value will be investigated. The optimal patch size $P \times Q$ is unclear and will also be studied; without loss of generality we assume $P = Q$.

The regularization parameter λ in (4) balances the matrix factorization error and the sparsity constraint on the elements of the matrix H. The larger the λ, the more weight is given to minimization of $\|H\|_{\text{sum}}$, while for small λ more weight is given to minimization of the factorization error. If $\lambda = 0$ then (4) reduces to the classical nonnegative matrix factorization problem.

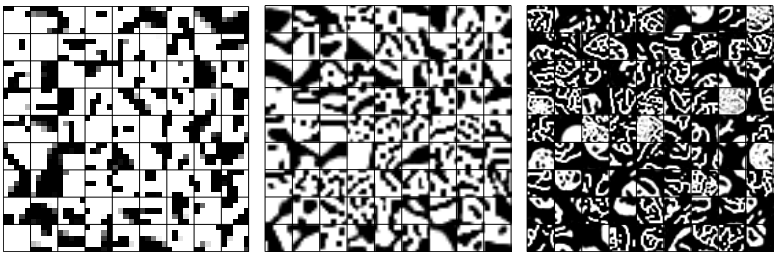
From the analysis of the upper bound on the regularization parameter λ in the Appendix, we know $\lambda \geq p$ implies $H = 0$; so λ can be varied in the interval $(0, p]$ to find dictionaries with different sparsity priors. Note that the scaling of the training images affects the scaling of the matrix H as well as the regularization parameter λ.

To evaluate the impact of the dictionary parameters, we use three different patch sizes (5×5, 10×10, and 20×20) and the number of dictionary elements s is chosen to be 2, 3, and 4 times the of the number of rows p in the dictionary D. We extract more than $50,000$ overlapping patches from the high-resolution image in Figure 2. For different combinations of patch sizes and number of dictionary elements we solve the dictionary learning problem (4).

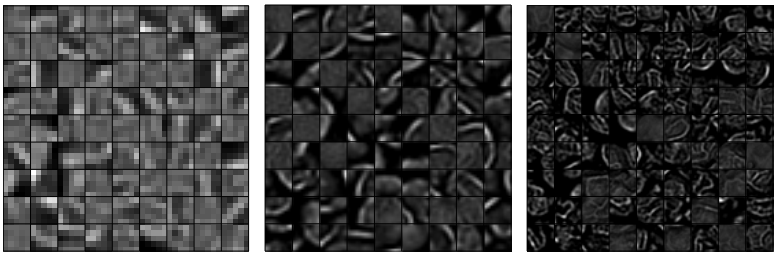
Figure 3 shows examples of such learned dictionaries, where columns of D are represented as images; we see that the penalty constraint $D \in \mathcal{D}_\infty$ gives rise to "binary looking" dictionary elements while $D \in \mathcal{D}_2$ results in dictionary elements that use the whole gray-scale range.

To evaluate the approximation error, i.e., the distance of the exact image x^{exact} to its projection on the cone C (6), we compute the solutions $\alpha_j^\star$ to the q approximation problems for all blocks $j = 1, 2, \ldots, q$ in x^{exact},

$$\min_{\alpha_j} \tfrac{1}{2} \left\| D\alpha_j - x_j^{\text{exact}} \right\|_2^2 \qquad \text{s.t.} \qquad \alpha_j \geq 0. \tag{9}$$



(a) 5×5, $s = 100$ (b) 10×10, $s = 300$ (c) 20×20, $s = 800$

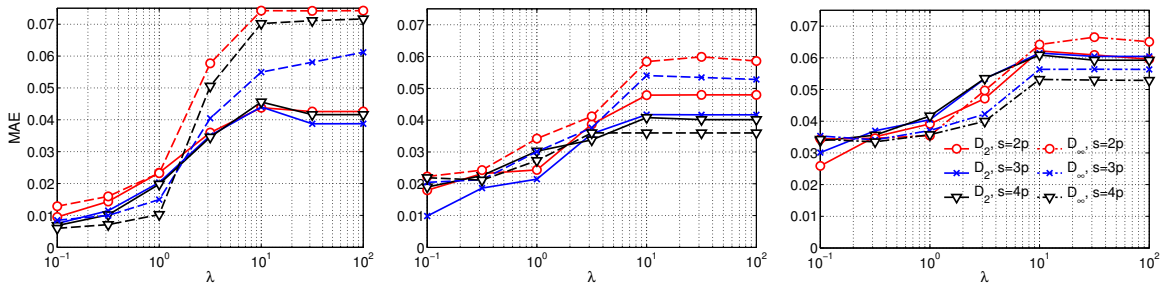


(d) 5×5, $s = 100$ (e) 10×10, $s = 300$ (f) 20×20, $s = 800$

Figure 3: Examples of dictionary elements. Top row: with the constraint $D \in \mathcal{D}_\infty$ the images appear as "binary looking." Bottom row: with the constraint $D \in \mathcal{D}_2$ the images appear to use the whole gray-scale range.

Then $P_C(x_j^{\text{exact}}) = D\alpha_j^\star$ is the best representation/approximation of the jth block in the cone. The mean approximation error (MAE) is then computed as

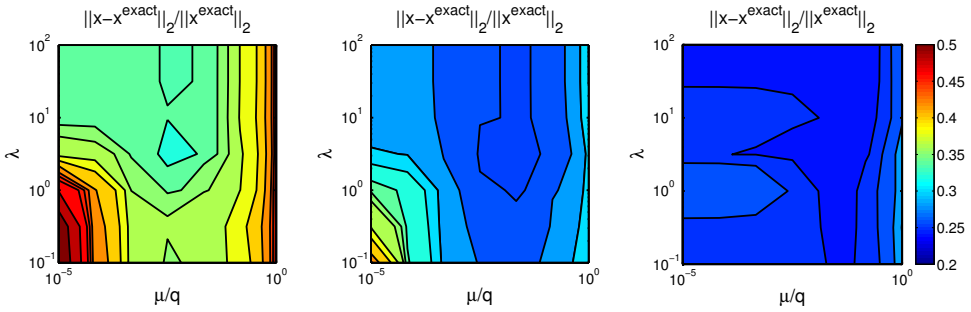
$$\text{MAE} = \frac{1}{q}\sum_{j=1}^{q}\frac{1}{\sqrt{p}}\left\|P_C(x_j^{\text{exact}}) - x_j^{\text{exact}}\right\|_2. \tag{10}$$



5×5 patches 10×10 patches 20×20 patches

Figure 4: Mean approximation errors (10) for both $D \in \mathcal{D}_\infty$ and $D \in \mathcal{D}_2$ with different patch sizes and different s.

The ability of the dictionary to represent features and textures from the training images, which determines how good reconstructions we are able to compute, depends on the regularization parameter λ, the patch size, and the number of dictionary elements. Figure 4 shows how the mean approximation error MAE (10) associated with the dictionary varies with patch size p, number of dictionary elements s, and regularization parameter λ. An advantage of larger patch sizes is that the variation of MAE with s and λ is less pronounced than for small patch sizes, so overall we tend to prefer larger patch sizes. In particular, for a large patch size we can use a smaller over-representation factor s/p than for a small patch size. As λ approaches p we have that $\|H\|_{\text{sum}}$ approaches 0, the dictionary D takes arbitrary values, and the approximation errors level off at a maximum value. Regarding the two different constraints $D \in \mathcal{D}_\infty$



(a) 5×5 patches (b) 10×10 patches (c) 20×20 patches

Figure 5: Contour plots of the reconstruction error RE (11) versus λ and μ/q.

and $D \in \mathcal{D}_2$ we do not see any big difference in the approximation errors for 10×10 and 20×20 patches; to limit the amount of results we now use $\mathcal{D}_2$.

The computational work depends on the patch size and the number of dictionary elements which, in turn, affects the approximation error: the larger the dictionary, the smaller the approximation error, but at a higher computational cost. We have found that a good trade-off between the computational work and the approximation error can be obtained by increasing the number of dictionary elements until the approximation error levels off.

4.3. Studies of the Reconstruction Stage

Here we evaluate the overall reconstruction framework including the effect of the reconstruction parameters as well as their connection to the dictionary learning parameter λ and the patch size.

We solve the reconstruction problem (8) using projection data based on the exact image given in Figure 2. We choose $N_p = 25$ uniformly distributed projection angles in $[0°, 180°]$. Hence the matrix A has dimensions $m = 7,050$ and $n = 40,000$, so the problem is highly underdetermined. We use the relative noise level $\|e\|_2/\|Ax^{\text{exact}}\|_2 = 0.01$. Moreover, we use 5×5, 10×10 and 20×20 patches and corresponding dictionary matrices $D^{(5)}$, $D^{(10)}$, and $D^{(20)}$ in $\mathcal{D}_2$ of size 25×100, 100×300, and 400×800, respectively. Examples of the dictionary elements are shown in the bottom row of Figure 3.

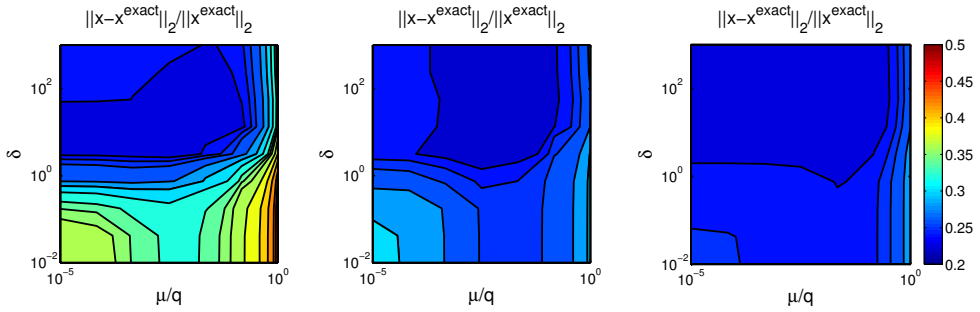
We first investigate the reconstruction's sensitivity to the choice of λ in the dictionary learning problem and the parameters μ and δ in the reconstruction problem. It follows from the optimality conditions of (8) that $\alpha^\star = 0$ is optimal when $\mu \geq \bar{\mu} = \frac{q}{m}\|(I \otimes D^T)\Pi^T A^T b\|_\infty$ and hence we choose $\mu \in [0, \bar{\mu}]$. Large values of μ refer to the case where the sparsity prior is strong and the solution is presented with too few dictionary elements. On the other hand if μ is small and a sufficient number of dictionary elements are included, the reconstruction error worsens only slightly when μ decreases. In the next subsection we show that we may, indeed, obtain reasonable reconstructions for $\mu = 0$.

To investigate the effect of regularization parameters λ and μ, we first perform experiments with $\delta = 0$ corresponding to no image prior. The quality of a solution x is evaluated by the *reconstruction error*

$$\text{RE} = \|x - x^{\text{exact}}\|_2/\|x^{\text{exact}}\|_2 \tag{11}$$

and Figure 5 shows contour plots of RE as a function of λ and μ/q. The reconstruction error is smaller for larger patch sizes, and also less dependent on the regularization parameter λ and the normalized regularization parameter μ/q. The smallest reconstruction errors are obtained in all dictionary sizes for $\lambda \approx 3$.

Let us now consider the reconstructions when $\delta > 0$ in order to reduce block artifacts. Figure 6 shows contour plots of the reconstruction error versus μ/q and δ, using a fixed $\lambda = 3.16$. It is no surprise that introducing δ acts as a regularizer that can significantly reduce blocking artifacts and thus improve the reconstruction. Sufficiently large values of δ yield smaller reconstruction errors. Consistent with the results from Figure 5, the reconstruction errors are smaller for 10×10 and 20×20 patch sizes than for 5×5 patches. For larger patch sizes (which allow for capturing more structure in the dictionary elements) the reconstruction error is quite insensitive to the choice of δ and μ. The contour plots in Figure 6 suggest that with our problem specification, we should choose $\delta \geq 1$.



(a) 5×5 patches (b) 10×10 patches (c) 20×20 patches

Figure 6: Contour plots of the reconstruction errors RE (11) versus μ/q and δ for a fixed $\lambda = 3.16$.

Finally, in Figure 7 we compare our reconstructions with those computed by means of filtered back projection (FBP), the algebraic reconstruction technique (ART), and TV regularization. We used the Shepp-Logan filter in "iradon." To be fair, the TV regularization parameter and the number of ART iterations were chosen to yield an optimal reconstruction.

- The FBP reconstruction contains the typical artifacts associated with this method for underdetermined problems, such as line structures.
- The ART reconstruction – although having about the same RE as our reconstruction – is blurry and contains artifacts such as circle structures and errors in the corners.
- The TV reconstruction has the typical "cartoonish" appearance of TV solutions and hence it fails to include most of the details associated with the texture; the edges of the pepper grains are distinct but geometrically somewhat un-smooth.
- Our reconstructions, while having about the same RE as the TV reconstruction, include more texture and some of the details from the exact image (but not all) are recovered, especially with $D^{(20)}$. Also the pepper grain edges resemble more the smooth edges from the exact image.

We conclude that our dictionary-based reconstruction method appears to have an edge over the other three methods.

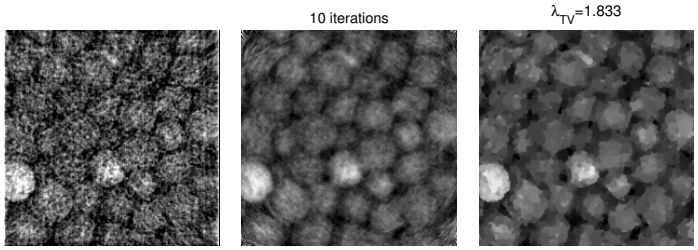
Our formulation in (8) enforces that the solution is an exact representation in the dictionary, and searching for solutions in the cone spanned by the dictionary elements is a strong assumption in the reconstruction formulation. In [33] we investigated this requirement experimentally and showed that relaxing the equality $x = \Pi\,(I \otimes D)$ does not give an advantage, i.e., approximating a solution by $x \approx \Pi\,(I \otimes D)\alpha$ and minimizing $\|x - \Pi\,(I \otimes D)\alpha\|_2$ does not improve the reconstruction quality, and one can compute a good reconstruction as a conic combination of the dictionary elements.

4.4. Simplifying the Computational Problem

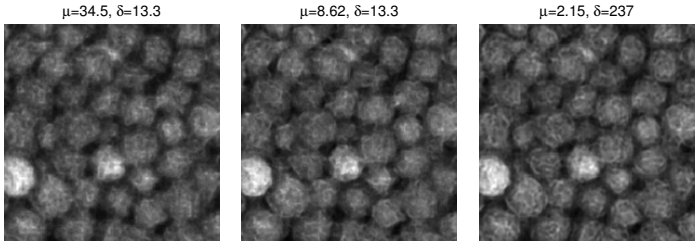
We have been working under the assumption that $\alpha \geq 0$ and that it is sparse. Imposing both non-negativity and a 1-norm constraint on the representation vector α are strong assumptions in the reconstruction formulation. If we drop the non-negativity constraint in the image reconstruction problem, then (8) takes the form of a constrained least squares problem:

$$\min_{\alpha} \frac{1}{2} \left\| \begin{pmatrix} \frac{1}{\sqrt{m}} A\,\Pi\,(I \otimes D) \\ \frac{\delta}{\sqrt{\ell}} L\,\Pi\,(I \otimes D) \end{pmatrix} \alpha - \begin{pmatrix} b \\ 0 \end{pmatrix} \right\|_2^2 \qquad \text{s.t.} \quad \|\alpha\|_1 \leq \gamma, \tag{12}$$

where $\gamma > 0$. Alternatively we can neglect the parameter μ. This is motivated by the plots in Figures 5 and 6 which suggest that for sufficiently large λ, δ and patch sizes, the reconstruction error is almost independent of μ as long as it



(a) FBP, RE = 0.481 (b) ART, RE = 0.225 (c) TV, RE = 0.214



(d) 5×5, RE = 0.224 (e) 10×10, RE = 0.220 (f) 20×20, RE = 0.226

Figure 7: Reconstructions for different patch sizes, with $D \in \mathcal{D}_2$ and $\lambda = 3.16$, compared with the FBP, ART and TV solutions. Note that in all our three reconstructions $\mu/q = 0.022$. RE denotes the reconstruction error (11).

is small. When $\mu = 0$ (8) reduces to a nonnegatively constrained least square problem:

$$\min_{\alpha} \frac{1}{2} \left\| \begin{pmatrix} \frac{1}{\sqrt{m}} A\,\Pi\,(I \otimes D) \\ \frac{\delta}{\sqrt{\ell}} L\,\Pi\,(I \otimes D) \end{pmatrix} \alpha - \begin{pmatrix} b \\ 0 \end{pmatrix} \right\|_2^2 \qquad \text{s.t.} \qquad \alpha \geq 0. \tag{13}$$

We use the same test problem with 25 projections and relative noise level 0.01 as in Section 4.3. We solve problem (12) for $D^{(10)} \in \mathcal{D}_2$, which resulted in the smallest reconstruction error when solving (8) (cf. Figure 7). Likewise we choose 10×10 and 20×20 patch sizes and $D^{(10)}, D^{(20)} \in \mathcal{D}_2$ to solve (13). Figures 8 and 9 show the respective reconstructions.

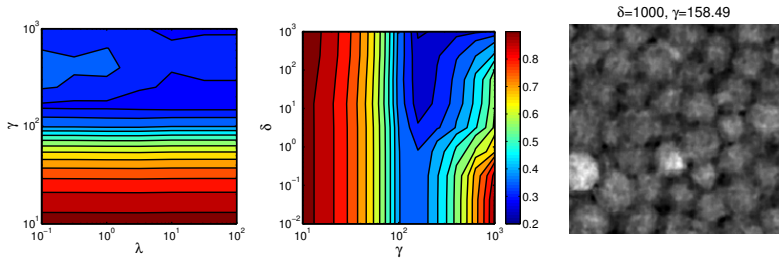


Figure 8: Contour plots of the reconstruction error RE for problem (12), similar to Figures 5 and 6. Left: RE versus λ and γ when $\delta = 0$. Middle: RE versus γ and δ with fixed $\lambda = 10$. Right: The best reconstruction with RE = 0.243.

There are two difficulties with the reconstructions computed via (12). The lack of a nonnegativity constraint on α can lead to negative pixel values in the reconstruction, and this is undesired because it is nonphysical and it leads to a larger reconstruction error . Also, as can be seen in Figure 8, the reconstruction is very sensitive to the choose of the regularization parameter γ; it must be sufficiently large to allow the solution to be represented with a sufficient number of dictionary elements, and it should be carefully chosen to provide an acceptable reconstruction.

The solution to problem (13) for a 20×20 patch size, compared to the solution shown in Figure 7, is not significantly worse both visually and in terms of reconstruction error. This suggests that using the dictionary obtained

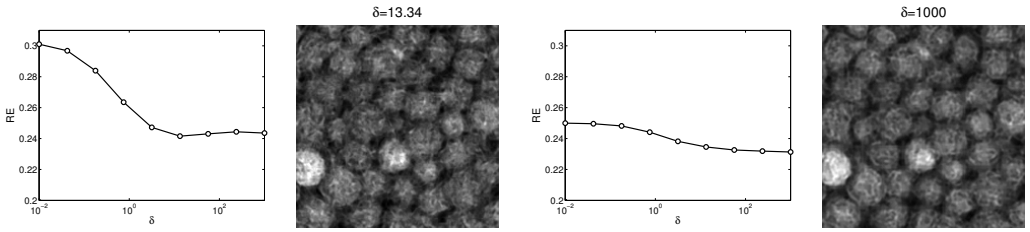


Figure 9: Plots of reconstruction error versus δ for problem (13), using fixed $\lambda = 3.16$ and $\mu = 0$, together with the best reconstructions with RE = 0.242 and RE = 0.231. Left and right parts correspond to patch sizes 10×10 and 20×20, respectively.

from (4) with a proper choice of λ and patch size and a nonnegatively constraint may be sufficient for the reconstruction problem, i.e., we can let $\mu = 0$. While this seems to simplify the problem – going from (8) to (13) – it does not significantly simplify the computational optimization problem, since the 1-norm constraint is handled by simple thresholding in the software; but it help us to get rid of a parameter in the reconstruction process. However, when the 1-norm constraint is omitted, additional care is necessary when choosing λ and the patch sizes to avoid introducing artifacts or noise.

4.5. Studies of Robustness

To further study the performance of our algorithm, in this section we consider reconstructions based on (8) with more noise in the data, and with projections within a limited range. The first two tests use 25 and 50 projections with uniform angular sampling in $[0°, 180°]$ and with relative noise level = 0.05, i.e., a higher noise level than above. For our highly underdetermined problems we know that both filtered back projection and algebraic iterative techniques give unsatisfactory solutions, and therefore we only compare our method with TV. As before the regularization parameters λ and μ are chosen from numerical experiments such that a solution with the smallest error is obtained.

The reconstructions are shown in the top and middle rows of Figure 10. The reconstruction errors are still similar across the methods. Again, the TV reconstructions have the characteristic "cartoonish" appearance while the dictionary-based reconstructions retain more the structure and texture but have other artifacts – especially for $N_p = 25$. We also note that these artifacts are different for the two different dictionaries.

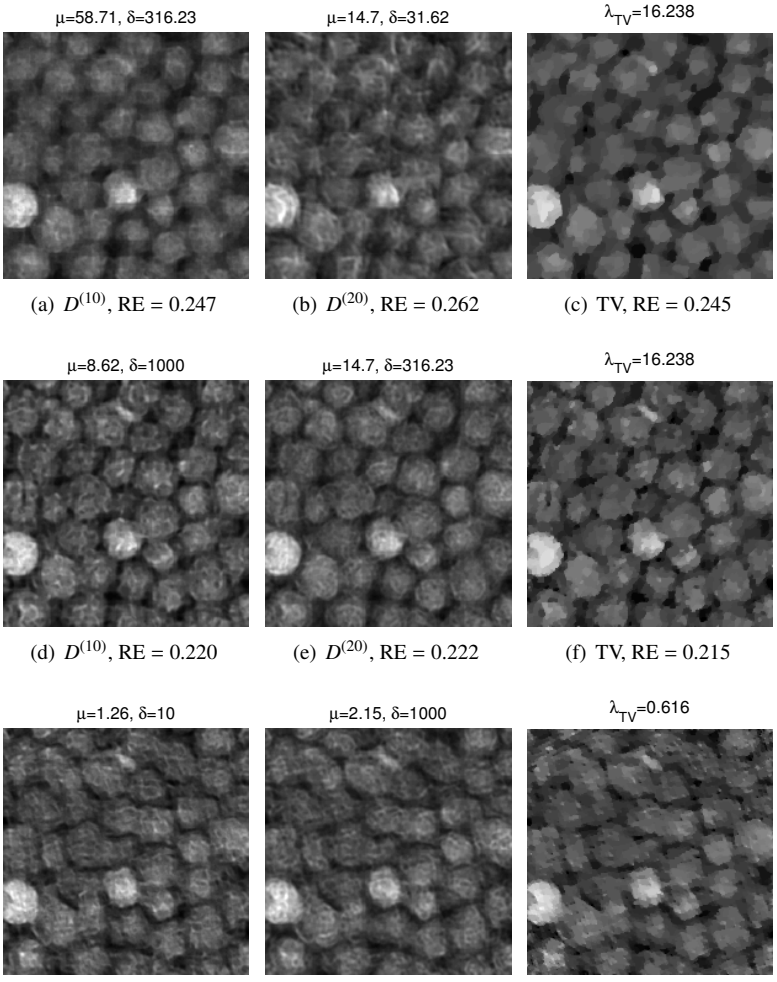
The third set uses 25 projections uniformly distributed in the limited range $[0°, 120°]$ and with relative noise level 0.01. In this case the TV reconstructions display additional artifacts related to the limited-angle situation, while such artifacts are somewhat less pronounced in the reconstructions by our algorithm.

In the numerical studies performed in this paper there is an underlying assumption that the scale and orientation of the training images are consistent with the unknown image. While this assumption is convenient for the studies performed here, it may not be entirely realistic. In a separate work [33] we therefore investigated the sensitivity and robustness of the reconstruction to variations of the scale and orientation in the training images, and we discuss algorithms to estimate the correct relative scale and orientation from the data (scale being the more difficult parameter to estimate).

4.6. A Large Test Case

We also verified our method on two larger test problems that simulate the analysis of microstructure in materials science. Almost all common metals and many ceramics are polycrystalline, i.e., they are composed of many small crystals or grains of varying size and orientation, and the variations in orientation can be random. It is of particular interest to study how the grain boundaries — the interfaces between grains — respond external stimuli such heat, stress or strain. Here we assume that priors of the grain structure are available in the form of training images.

The simulated data was computed using images of steel and zirconium grains. The steel microstructure image from [42] is of dimensions 900×1280 and the zirconium grain image (produced by a scanning electron microscope) is 760×1020. More than $50,000$ patches are extracted from these images to learn dictionaries $D^{(20)} \in \mathcal{D}_2, \mathcal{D}_\infty$ of size 400×800. To avoid doing inverse crime, we obtain the exact images of dimensions 520×520 by first rotating the high-resolution image and then extracting the exact image. The high-resolution images and the exact images are shown in Figure 11.



(a) $D^{(10)}$, RE = 0.247 (b) $D^{(20)}$, RE = 0.262 (c) TV, RE = 0.245

(d) $D^{(10)}$, RE = 0.220 (e) $D^{(20)}$, RE = 0.222 (f) TV, RE = 0.215

(g) $D^{(10)}$, RE = 0.255 (h) $D^{(20)}$, RE = 0.261 (i) TV, RE = 0.246

Figure 10: The left and middle columns show our reconstructions with $\lambda = 3.16$ using $D^{(10)}$ and $D^{(20)}$, respectively; the right column shows the TV reconstructions. Top and middle rows: $N_p = 25$ and $N_p = 50$ projections in $[0°, 180°]$ and relative noise level 0.05. Bottom row: $N_p = 25$ projections in $[0°, 120°]$ and relative noise level 0.01.

We consider a parallel-beam tomographic scenario with $N_{\mathrm{p}} = 50$ projections corresponding to 50 uniformly distributed projections in $[0°, 180°]$, leading to $m = 36,750$ measurements. We add Gaussian white noise with relative noise level 0.01 and compute reconstructions by our method as well as the TV method; these reconstruction are shown in Figure 12. All regularization parameters were chosen to give the best reconstruction as measured by the RE, and we note that the reconstruction errors are dominated by the error coming from the regularization of the noisy data; the approximation errors $\|P_C(x^{\mathrm{exact}}) - x^{\mathrm{exact}}\|_2 / \|x^{\mathrm{exact}}\|_2$ are of the order 0.03 and 0.05 for the steel and zirconium images, respectively.

As expected, the TV reconstructions exhibit "cartoonish" artifacts, and for the steel grains the black interfaces tend to be too thick and they are not so well resolved. Our method, for both $\mathcal{D}_2$ and $\mathcal{D}_\infty$, recovers better the grain interfaces that are of interest here. We obtain the sharpest interfaces for $\mathcal{D}_\infty$ but some small black "dots" have appeared which are not present for $\mathcal{D}_2$; in both cases the images are suited for postprocessing via image analysis.

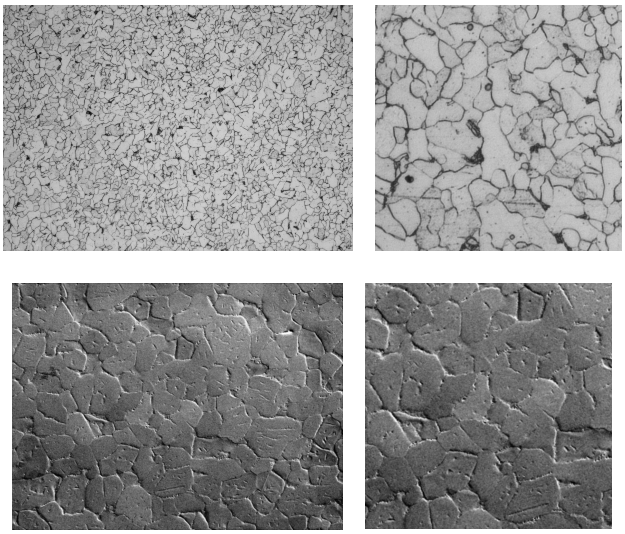
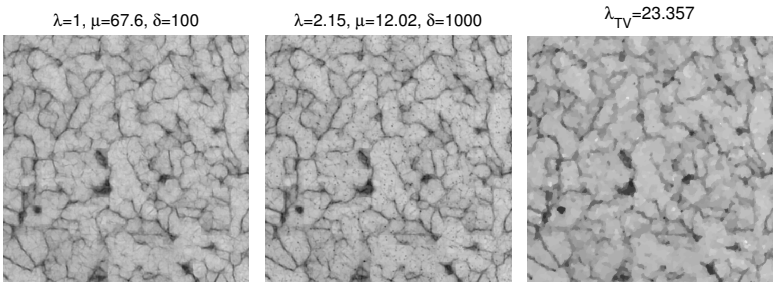


Figure 11: Left: high-resolution images of steel micro-structure [42] (top) and zirconium grains (bottom) used to generate the training images. Right: the corresponding exact images of size 520×520.



(a) $D^{20} \in \mathcal{D}_2$, RE = 0.095 (b) $D^{20} \in \mathcal{D}_\infty$, RE = 0.096 (c) TV, RE = 0.099

(d) $D^{20} \in \mathcal{D}_2$, RE = 0.146 (e) $D^{20} \in \mathcal{D}_\infty$, RE = 0.158 (f) TV, RE = 0.137

Figure 12: Reconstructions of the 520×520 images by our method (left and middle) and by the TV method (right). Top: steel microstructure. Bottom: zirconium grains.

4.7. The Impact of Wrong Training Images

We finish our numerical experiments with an artificial example that clearly illustrates the impact of a wrong set of training images. Assume that we again use the $\mathcal{D}_2$-dictionary image patches from the peppers training image, but now we reconstruct an image of matches (that have no resemblance to the peppers). All parameters are identical to those from Sections 4.2–4.3 and we use a patch size of 20×20. The exact image and the "best" reconstruction are shown in

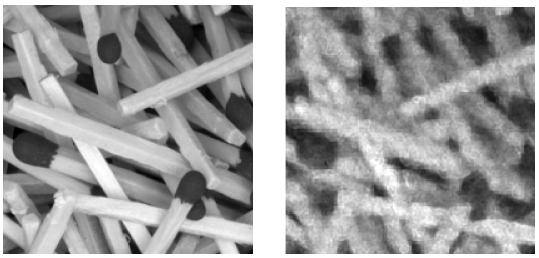


Figure 13: Left: the exact image. Right: the "best" reconstruction based on a dictionary created from the peppers training image.

Fig. 13; it is evident that the dictionary forms a strong prior with the result that the reconstruction is so influence by the peppers prior that it is very bad reconstruction of the matches.

5. Conclusions

We describe and examine an algorithm that incorporates training images as priors in computed tomography (CT) reconstruction problems. This type of priors can be useful in low-dose CT where we are faced with underdetermined systems of equations, and our numerical experiments focus on such problems.

Our algorithm has two stages. In the first stage we compute a learned dictionary from a set of training images using a regularized nonnegative matrix factorization (NMF). In the second stage, via a regularized least squares fit we compute a nonnegative reconstruction lying in the cone defined by the dictionary elements; the reconstruction is sparse with respect to the dictionary. Hence, regularization is obtained by enforcing that the reconstruction is within the range of the dictionary elements and by the sparsity constraint. An extension of this algorithm to a tensor formulation, that avoids reshaping of image patches to vectors, is presented in [34].

Our algorithm works with non-overlapping image patches; the same dictionary is used for all patches, and we are able to minimize blocking artifacts by an additional regularization term. This reduces the computational complexity, compared to all other proposed algorithms that apply a dictionary-based regularization based on overlapping patches around every pixel in the image.

Our algorithm includes several regularization parameters. In the first stage a parameter is used to control the sparsity in the NMF, and in the second stage we use one parameter to control the sparsity of the representation in the dictionary, and another parameter to avoid blocking artifacts. We perform a series of numerical experiments with noisy data and without committing inverse crime, where we demonstrate the interplay between these parameters and the computed reconstructions, and we show that the reconstructions are not very sensitive to these parameters. Further work is needed to develop automatic parameter choice algorithms.

We conclude that training images can be useful as a strong prior for regularization of low-dose CT problems, through a sparse representation in a nonnegative dictionary learned from the training images. Our reconstructions are (not surprisingly) superior to those computed by classical methods such as filtered back projection and algebraic iterative methods, and they are competitive with total variation (TV) reconstructions. Specifically, in our test problems our algorithm tends to be able to include more texture and also produces edges whose location is more correct.

Acknowledgments

The authors would like to thank Prof. Samuli Siltanen from Univ. of Helsinki for providing the high-resolution image of the peppers, and Dr. Hamidreza Abdolvand from Univ. of Oxford for providing the zirconium image.

Appendix A. The Dictionary Learning Algorithm

Recall that the dictionary learning problem (4) is non-convex, and hence it is too costly to solve it globally. We will therefore optimize locally by applying the Alternating Direction Method of Multipliers (ADMM) method [4]to

the following reformulation of (4)

$$\begin{array}{ll} \text{minimize}_{D,H} & \frac{1}{2}\left\|Y - UV\right\|_{\mathrm{F}}^2 + \lambda \left\|H\right\|_{\mathrm{sum}} + I_{\mathbb{R}_+^{s\times t}}(H) + I_{\mathcal{D}}(D) \\ \text{subject to} & D = U,\ H = V, \end{array} \tag{A.1}$$

where $U \in \mathbb{R}^{p\times s}$ and $V \in \mathbb{R}^{s\times t}$ are auxiliary variables that are introduced in order to make the ADMM-updates separable and hence cheap. The augmented Lagrangian associated with (A.1) can be expressed as

$$\begin{aligned} L_\rho(D, H, U, V, \Lambda, \Pi) = \frac{1}{2}\|Y - UV\|_{\mathrm{F}}^2 + \lambda \left\|H\right\|_{\mathrm{sum}} + I_{\mathbb{R}_+^{s\times t}}(H) + I_{\mathcal{D}}(D) \\ + \mathrm{Tr}(\Lambda^T(D - U)) + \mathrm{Tr}(\Pi^T(H - V)) \\ + \frac{\rho}{2}\|D - U\|_{\mathrm{F}}^2 + \frac{\rho}{2}\|H - V\|_{\mathrm{F}}^2 \end{aligned} \tag{A.2}$$

where $\Lambda \in \mathbb{R}^{p\times s}$ and $\Pi \in \mathbb{R}^{s\times l}$ are Lagrange multipliers, and ρ is a fixed positive penalty parameter which can be chosen prior to the learning process. If we partition the variables into two blocks (D, V) and (H, U) and apply ADMM to (A.1), we obtain an algorithm where each iteration involves the following three steps: (i) minimize L_ρ jointly over D and V; (ii) minimize L_ρ jointly over H and U; and (iii) update the dual variables Λ and Π by taking a gradient-ascent step. Since L_ρ is separable in D and V, step (i) can be expressed as two separate updates

$$D_{k+1} = \min_{D\in\mathcal{D}} L_\rho(D, H_k, U_k, V_k, \Lambda_k, \Pi_k) = P_{\mathcal{D}}(U_k - \rho^{-1}\Lambda_k) \tag{A.3a}$$

$$\begin{aligned} V_{k+1} &= \min_{V} L_\rho(D_k, H_k, U_k, V, \Lambda_k, \Pi_k) \\ &= (U_k^T U_k + \rho I)^{-1}(U_k^T Y + \Pi_k + \rho H_k) \end{aligned} \tag{A.3b}$$

where $P_{\mathcal{D}}(\cdot)$ is the projection onto the set $\mathcal{D}$. Similarly, L_ρ is also separable in H and U, so step (ii) can be written as

$$\begin{aligned} H_{k+1} &= \min_{H\in\mathbb{R}_+^{s\times t}} L_\rho(D_{k+1}, H, U_k, V_{k+1}, \Lambda_k, \Pi_k) \\ &= P_{\mathbb{R}_+^{s\times t}}(\mathcal{S}_{\lambda/\rho}(V_{k+1} - \rho^{-1}\Pi_k)) \end{aligned} \tag{A.3c}$$

$$\begin{aligned} U_{k+1} &= \min_{U} L_\rho(D_{k+1}, H_k, U, V_{k+1}, \Lambda_k, \Pi_k) \\ &= (YV_{k+1}^T + \Lambda_k + \rho D_{k+1})(V_{k+1}V_{k+1}^T + \rho I)^{-1} \end{aligned} \tag{A.3d}$$

where $\mathcal{S}_{\lambda/\rho}$ denotes an entrywise soft-thresholding operator, and $P_{\mathbb{R}_+^{s\times t}}(\cdot)$ is the projection onto the non-negative orthant. Finally, the dual variable updates in step (iii) are given by

$$\Lambda_{k+1} = \Lambda_n + \rho(D_{k+1} - U_{k+1}) \tag{A.3e}$$

$$\Pi_{k+1} = \Pi_k + \rho(H_{k+1} - V_{k+1}). \tag{A.3f}$$

The projection onto the set $\mathcal{D}_\infty$ is an element-wise projection onto the interval $[0, 1]$ and hence easy to compute. However, the projection onto $\mathcal{D}_2$ does not have a closed form solution, so we compute it iteratively using Dykstra's alternating projection algorithm.

The convergence properties of ADMM when applied to non-convex problems of the form (A.1) have been studied by e.g. [40]. They show that whenever the sequence of iterates produced by (A.3) converges, the limit satisfies the the KKT-conditions (i.e., the first-order necessary conditions for optimality) which can be expressed as

$$D = U, \quad H = V,$$

$$\Lambda = -(Y - DH)H^T, \quad \Pi = -D^T(Y - DH),$$

$$-\Lambda \in \partial\Phi_{\mathrm{dic}}(D), \quad -\Pi \in \partial\Phi_{\mathrm{rep}}(H),$$

where ∂ denotes the subdifferential operator. The convergence result is somewhat weak, but empirical evidence suggests that applying ADMM to non-convex problems often works well in practice [4]. It is interesting to note that

the point $D = U = 0$ and $H = V = 0$ satisfies the KKT-conditions, and although it is a stationary point, it is clearly not a local minima. For this reason, we avoid initializing with zeros. We initialize U with some of the images from the training set, and we set $V = [I\ 0]$ (i.e., the leading s columns of V is the identity matrix).

The KKT-conditions can be used to formulate stopping criteria. We use the following conditions

$$\frac{\|D - U\|_{\max}}{\max(1, \|D\|_{\max})} \leq \epsilon \quad \wedge \quad \frac{\|H - V\|_{\max}}{\max(1, \|H\|_{\max})} \leq \epsilon \tag{A.4a}$$

$$\frac{\|\Pi - D^T(DH - Y)\|_{\max}}{\max(1, \|\Pi\|_{\max})} \leq \epsilon \quad \wedge \quad \frac{\|\Lambda - (DH - Y)H^T\|_{\infty}}{\max(1, \|\Lambda\|_{\max})} \leq \epsilon \tag{A.4b}$$

where $\epsilon > 0$ is a given tolerance.

The KKT-conditions can also be used to derive an upper bound $\bar{\lambda}$ for the regularization parameter λ. It follows from the optimality conditions that for $H = 0_{s\times t}$, $\Pi = -D^{\mathrm{T}}Y$ and hence for some $\bar{\lambda}$ and all $D \in \mathcal{D}$ we have

$$D^{\mathrm{T}}Y \in \bar{\lambda}\, \partial \|0_{s\times t}\|_{\mathrm{sum}},$$

i.e., $H = 0$ satisfies the KKT-conditions for all $\lambda \geq \bar{\lambda}$. Thus, if Y is scaled such that all entries in Y are between 0 and 1, then the upper bound $\bar{\lambda} = p$ can be used for both dictionaries since

$$\sup_{D\in\mathcal{D}_2} \|D^{\mathrm{T}}Y\|_{\max} = \max_{j=1,\ldots,t} \sqrt{p}\|Ye_j\|_2 \leq p \qquad \text{and} \qquad \sup_{D\in\mathcal{D}_\infty} \|D^{\mathrm{T}}Y\|_{\max} = \max_{j=1,\ldots,t} \|Ye_j\|_1 \leq p$$

which implies that $D^{\mathrm{T}}Y \in \bar{\lambda}\, \partial\Phi_{\mathrm{rep}}(0_{s\times t})$ for all $D \in \mathcal{D}$.

References

[1] M. Aharon, M. Elad, and A. Bruckstein, *K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation*, IEEE Trans. Signal Process., **54** (2006), 4311–4322.

[2] S. Becker, E. J. Candès, and M. Grant, *Templates for convex cone problems with applications to sparse signal recovery*, Math. Prog. Comp., **3** (2011), 165–218.

[3] J. Bian, J. H. Siewerdsen, X. Han, E. Y. Sidky, J. L. Prince, C. A. Pelizzari, and X. Pan, *Evaluation of sparse-view reconstruction from flat-panel-detector cone-beam CT*, Phys. Med. Biol., **55** (2010), 6575–6599.

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Foundations and Trends in Machine Learning, **3** (2011), 1–122.

[5] A. M. Bruckstein, D. L. Donoho, and M. Elad, *From sparse solutions of systems of equations to sparse modeling of signals and images*, SIAM Review, **51** (2009), 34–81.

[6] T. M. Buzug, *Computed tomography: From photon statistics to modern cone-beam CT*, Springer, 2010.

[7] Y. Chen, Xi. Yin, L. Shi, H. Shu, L. Luo, J. L. Coatrieux, and C. Toumoulin, *Improving abdomen tumor low-dose CT images using a fast dictionary learning based processing*, Phys. Med. Biol., **58** (2013), 5803–5820.

[8] S. Chen, H. Liu, P. Shi, and Y. Chen, *Sparse representation and dictionary learning penalized image reconstruction for positron emission tomography*, Phys. Med. Biol., **60** [2015), 807–823.

[9] M. Elad, *Sparse and redundant representations, from theory to applications in signal and image processing*, Springer, New York, 2010.

[10] M. Elad and M. Aharon, *Image denoising via sparse and redundant representations over learned dictionaries* IEEE Trans. Image Process., **15** (2006), 3736–3745.

[11] K. Engan, S. O. Aase, and J. H. Husøy, *Multi-frame compression: theory and design*, EURASIP Signal Process., **80** (2000), 2121–2140.

[12] V. Etter, I. Jovanović, and M. Vetterli, *Use of learned dictionaries in tomographic reconstruction*, Proc. SPIE **8138**, Wavelets and Sparsity XIV (2011) 81381C.

[13] P. C. Hansen, *Discrete Inverse Problems – Insight and Algorithms*, SIAM, Philadelphia, 2010.

[14] P. C. Hansen and M. Saxild-Hansen, *AIR Tools – A MATLAB package of algebraic iterative reconstruction methods*, J. Comput. Appl. Math., **236** (2012), 2167-2178.

[15] P. O. Hoyer, *Non-negative matrix factorization with sparseness constraints*, J. Mach. Learn. Res., **5** (2004), 1457–1469.

[16] Y. Huang, J. Paisley, Q. Lin, X. Ding, X. Fu, and X.-P. Zhang. *Bayesian nonparametric dictionary learning for compressed sensing MRI*, IEEE Trans. Image Process., **23** (2014), Iss. 12.

[17] T. L. Jensen, J. H. Jørgensen, P. C. Hansen, and S. H. Jensen, *Implementation of an optimal first-order method for strongly convex total variation regularization*, BIT, **52** (2011), 329–356.

[18] A. Coates and A. Y. Ng, Learning feature representations with K-means, *Neural Networks: Tricks of the Trade*, (Eds. G. Montavon, G. B. Orr, K.-R. Müller), 2nd edn, Springer LNCS 7700, (2012), 561–580.

[19] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. Lee, and T. J. Sejnowski, *Dictionary learning algorithms for sparse representation*, Neural Comp., **15** (2003), 349–396.

[20] P. Kuchment, *The Radon Transform and Medical Imaging*, SIAM, Philadelphia, 2014.

[21] S. J. LaRoque, E. Y. Sidky, and Xi. Pan, *Accurate image reconstruction from few-view and limited-angle data in diffraction tomography*, J. Opt. Soc. Am. A Opt. Image Sci. Vis., **25** (2008), 1772–1782.

[22] D. D. Lee and H. S. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature, **401** (1999), 788–791.

[23] M.S.Lewicki and T.J. Sejnowski, *Learning overcomplete representations*, Neural Comp., **12** (2000), 337–365.

[24] Sh. Li, L. Fang, and H. Yin, *An efficient dictionary learning algorithm and its application to 3-D medical image denoising*, IEEE Trans. Biomed. Eng., **59** (2012), 417–427.

[25] Q. Liu, D. Liang, Y. Song, J. Luo, Y. Zhu, and W. Li, *Augmented Lagrangian-based sparse representation method with dictionary updating for image deblurring*, SIAM J. Imaging Sci., **6** (2013), 1689–1718.

[26] B. Liu, H. Yu, S.S. Verbridge, L. Sun, and G. Wang, *Dictionary learning-based reconstruction method for electron tomography*, Scanning, **36** (2014), 377–383.

[27] J. Mairal, G. Sapiro, and M. Elad, *Learning multiscale sparse representations for image and video restoration*, SIAM Multiscale Model. Simul., **7** (2008), 214–241.

[28] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, *Online learning for matrix factorization and sparse coding*, J. Mach. Learn. Res., **11** (2010), 19–60.

[29] A. Mirone, E. Brun, and P. Coan, *A dictionary learning approach with overlap for the low dose computed tomography reconstruction and its vectorial application to differential phase tomography*, PLOS ONE, **9** (2014), No. 12.

[30] J. L. Mueller and S. Siltanen, *Linear and Nonlinear Inverse Problems with Practical Applications*, SIAM, Philadephia, 2012.

[31] B. A. Olshausen and D. J. Field, *Emergence of simple-cell receptive field properties by learning a sparse code for natural images*, Nature, **381** (1996), 607–609.

[32] S. Ravishankar, and Y. Bresler, *MR image reconstruction from highly undersampled k-space data by dictionary learning*, IEEE Trans. Med. Imag., **30** (2011), 1028–1041.

[33] S. Soltani, *Studies of Sensitivity in the Dictionary Learning Approach to Computed Tomography: Simplifying the Reconstruction Problem, Rotation, and Scale*, Technical Report 2015-4, DTU Compute, Technical University of Denmark, 2015. http://orbit.dtu.dk/fedora/objects/orbit:140904/datastreams/file_112138797/content

[34] S. Soltani, M. E. Kilmer, and P. C. Hansen, *A tensor-based dictionary learning approach to tomographic image reconstruction*, BIT Numer. Math. (2016); DOI: 10.1007/s10543-016-0607-z.

[35] D. Strong and T. Chan, *Edge-preserving and scale-dependent properties of total variation regularization*, Inverse Problems, **19** (2003), S165–S187.

[36] R. Tibshirani, *Regression shrinkage and selection via the lasso*, J. R. Stat. Soc. Ser. B Stat. Methodol., **58** (1996), 267–288.

[37] J. A. Tropp and S. J. Wright, *Computational methods for sparse solution of linear inverse problems*, Proc. IEEE, **98** (2010), 948–958.

[38] J. Velikina, Sh. Leng, and G. H. Chen, *Limited view angle tomographic image reconstruction via total variation minimization*, Proc. SPIE **6510**, *Medical Imaging 2007: Physics of Medical Imaging*.

[39] Q. Xu, H. Yu, X. Mou, L. Zhang, J. Hsieh, and G. Wang, *Low-dose X-ray CT reconstruction via dictionary learning*, IEEE Trans. Med. Imag., **31** (2012), 1682–1697.

[40] Y. Xu, W. Yin, Z. Wen and Y. Zhang, *An alternating direction algorithm for matrix completion with nonnegative factors*, Front. Math. China, **7** (2012), 365–384.

[41] B. Zhao, H. Ding, Y. Lu, G. Wang, J. Zhao, and S. Molloi. *Dual-dictionary learning-based iterative image reconstruction for spectral computed tomography application*, Phys. Med. Biol., **57** (2012) Iss. 24.

[42] www.one-eighty-degrees.com/service/microstructural-investigations.