Joint CT Reconstruction and Segmentation with Discriminative Dictionary Learning

Dong, Yiqiu; Hansen, Per Christian; Kjer, Hans Martin

Published in:
Ieee Transactions on Computational Imaging

Link to article, DOI:
10.1109/TCI.2018.2858139

Publication date:
2018

Document Version
Peer reviewed version

Joint CT Reconstruction and Segmentation with Discriminative Dictionary Learning

Yiqiu Dong, Per Christian Hansen, and Hans Martin Kjer

***Abstract*—We present a novel algorithm for Computed Tomography (CT) that simultaneously computes a reconstruction and a corresponding segmentation. Our algorithm uses learned dictionaries for both the reconstruction and the segmentation, constructed via discriminative dictionary learning using a set of corresponding images and segmentations. We give a detailed description of the implementation of our algorithm, and computer simulations demonstrate that our method provides better results than the other SRS or dictionary-based methods, especially when there are not sufficient projections. Moreover, due to the regularization, the segmentations from our method has more smooth class interfaces.**

***Index Terms*—Tomographic reconstruction, segmentation, regularization, learned dictionaries, numerical optimization.**

I. Introduction

Image reconstruction problems in Computed Tomography (CT), and similar inverse problems, cannot be properly solved without regularization: prior knowledge about the solution must be incorporated into the reconstruction model [1]. Sometimes these priors are generic in nature, e.g., they express knowledge of the smoothness of the solution; but such priors fail to incorporate more specific knowledge about the solution and how it is further used in the data analysis process. For example, we often need to perform a segmentation, e.g., to separate an object from the background [2] or to identify specific objects or regions [3], and this has many applications in medical imaging and in non-destructive testing in materials science.

Traditionally, image reconstruction and segmentation are performed as two separate steps – even though reconstruction errors may propagate into misclassifications. More recent techniques perform the two steps simultaneously through a joint problem formulation, called Simultaneous Reconstruction and Segmentation (SRS), see, e.g., [4], [5], [6], [7], [8]. These methods have a potential advantage because the segmentation acts as a regularizer of the inverse problem; a potential disadvantage is that they may need more algorithm parameters.

In this work the regularization of the tomography problem is achieved through the use of training images that the computed image must resemble; this approach can handle priors that cannot be formulated in closed form. The underlying idea is to express the computed image as a sparse representation in terms of elements of a dictionary learned from the training images [9], [10], [11].

All three authors are with the Department of Applied Mathematics and Computer Science, Technical University of Denmark, Kgs. Lyngby, Denmark (e-mail: yido@dtu.dk, pcha@dtu.dk, hmkj@dtu.dk).

To avoid the computational effort from dealing with full large-size training images, we work only with patches from the training images as well as the reconstructed image. We follow the convention that every $p \times p$ image patch with $P = p^2$ pixels is represented by a $P \times 1$ vector, and we collect the N_{T} training image patch-vectors in a $P \times N_{\mathrm{T}}$ matrix $\boldsymbol{Y}$ (an alternative is to use a tensor formulation [12] but in this work we use the matrix formalism).

From the training images we computed a learned dictionary, which is a matrix $\boldsymbol{D}$ whose N_{D} columns, called the dictionary elements or atoms, are vectorized image patches that form the basis for our reconstruction. The dictionary is constructed such that sparse linear combinations of the dictionary elements represent the training images well. Hence, we compute the two matrices $\boldsymbol{D} \in \mathbb{R}^{P \times N_{\mathrm{D}}}$ and $\boldsymbol{H} \in \mathbb{R}^{N_{\mathrm{D}} \times N_{\mathrm{T}}}$ that solve the sparse coding problem

$$\min_{\boldsymbol{D},\boldsymbol{H}} \|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{H}\|_{\mathrm{F}}^2 + \gamma \left\|\mathtt{vec}(\boldsymbol{H})\right\|_1 . \tag{1}$$

Here, $\boldsymbol{H}$ is a matrix where each column represents the sparse approximation coefficients for the corresponding training image patch, $\mathtt{vec}(\boldsymbol{H})$ organizes the elements of $\boldsymbol{H}$ into a vector, and γ controls the degree of sparsity. The matrices $\boldsymbol{D}$ and $\boldsymbol{H}$ are obviously not uniquely determined, and different algorithms give different solution; see, e.g., [13], [14] for details.

Although patch-based methods cannot explore the global structures, regardless of the size of training data and dictionaries, dictionaries especially with small patches are indeed useful for handling edges and local textures. Dictionaries can be trained for other purposes such as classification, labelling, feature representation or artifact removal [15], [16], [17], [18], [19], and we focus on *discriminative dictionaries* that are trained for segmentation; see [20] for an application in image segmentation. Specifically, we use learned dictionaries for both the reconstruction and the segmentation in an SRS model. Our work can be seen as a merge and extension of [10], [12] and [20], and to our knowledge no-one has yet proposed to incorporate a discriminative dictionary into a model for tomographic SRS.

Our paper is organized as follows. In Section II we formulate our SRS model, and in Section III we describe the computational details of our algorithm. Section IV defines our numerical experiments, and in Section V we present and discuss the results. Conclusions are made in Section VI.

We assume familiarity with the terminology of CT imaging and the mathematical basics of tomographic reconstruction [21]. Throughout the paper we use the following notation:

N_C	number of classes in the segmentation
N_D	number of dictionary elements
N_P	number of patches in the reconstructed image
N_T	number of training image patches
n_p	number of CT projection angles
n_d	number of CT detector pixels

Upper and lower case boldface denote matrices and vectors, respectively.

II. The Proposed Method

Our method consists of two separate stages. In the first stage, a discriminative dictionary is learnt from problem-specific training data, using procedures established in other studies and extending the formulations from [16], [20]. In the second stage we introduce a new SRS model for CT image reconstruction and segmentation, using the learnt discriminative dictionary as the prior.

A. Discriminative Dictionary Learning

We assume that we have access to N_T high-quality $p \times p$ training image patches that represent the object under study, and that these images have been segmented such that each pixel is associated with one of N_C classes. In this work, the training image patches are extracted from random locations of a much larger training image.

Let $\boldsymbol{y}_i \in \mathbb{R}^P$ denote the ith vectorized training image patch. Each pixel of this patch is associated with a binary label vector $(0, \ldots, 0, 1, 0, \ldots, 0)^T \in \mathbb{R}^{N_\mathrm{C}}$, where the non-zero entry identifies the class label. We stack these label vectors into a single vector and permute the elements, such that the resulting vector $\boldsymbol{z}_i \in \mathbb{R}^{(PN_\mathrm{C})}$ has N_C blocks, each of length P, corresponding to the N_C classes; see Figure 1. (Note that it is possible to consider other strategies for labelling the patches; in [16] the entire patch is associated only with a single label.)

Given the N_T training image patch-vectors $\boldsymbol{y}_i$ and the corresponding class-label vectors $\boldsymbol{z}_i$, we organize these training data into two matrices (as illustrated in Figure 1):

- $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{N_\mathrm{T}}] \in \mathbb{R}^{P \times N_\mathrm{T}}$, the matrix of the training data patches.
- $\boldsymbol{Z} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{N_\mathrm{T}}] \in \mathbb{R}^{(PN_\mathrm{C}) \times N_\mathrm{T}}$, the matrix of the corresponding pixel classifications.

The task is then to *simultaneously* learn the image and class dictionaries $\boldsymbol{D} \in \mathbb{R}^{P \times N_\mathrm{D}}$ and $\boldsymbol{W} \in \mathbb{R}^{(PN_\mathrm{C}) \times N_\mathrm{D}}$, through the joint sparse coding problem:

$$\min_{\boldsymbol{D},\boldsymbol{W},\boldsymbol{H}} \|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{H}\|_\mathrm{F}^2 + \lambda^2 \|\boldsymbol{Z} - \boldsymbol{W}\boldsymbol{H}\|_\mathrm{F}^2 + \gamma \|\mathtt{vec}(\boldsymbol{H})\|_1 , \tag{2}$$

where λ defines the weighting of the grouping into classes and γ controls the sparsity. An illustration of a learned discriminative dictionary $\{\boldsymbol{D}, \boldsymbol{W}\}$ is given in Figure 1.

The structure of $\boldsymbol{W}$ follows that of $\boldsymbol{Z}$, and hence $\boldsymbol{W}$ can be partitioned into N_C blocks of sub-matrices $\boldsymbol{W}_k \in \mathbb{R}^{P \times N_\mathrm{D}}$, each corresponding to one of the N_C classes:

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_1 \\ \boldsymbol{W}_2 \\ \vdots \\ \boldsymbol{W}_{N_\mathrm{C}} \end{bmatrix} . \tag{3}$$

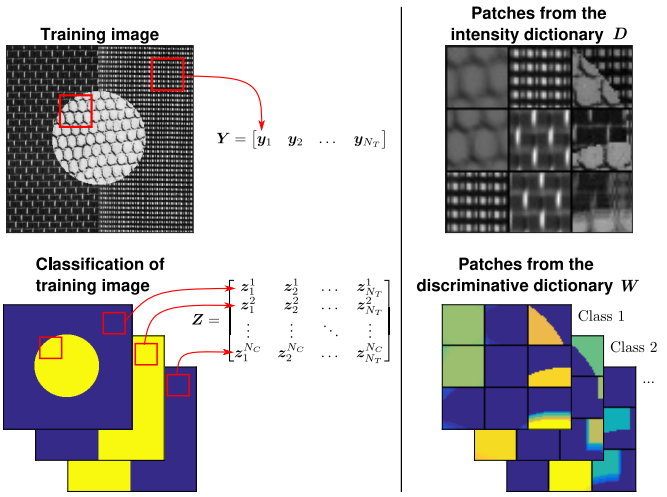

Fig. 1: Left: extraction of patches from a large training image; intensity patches are arranged as columns $\boldsymbol{y}_i$ of $\boldsymbol{Y}$; labels of the same patches are arranged as columns $\boldsymbol{z}_i$ of $\boldsymbol{Z}$ where sub-vector $\boldsymbol{z}_i^j \in \mathbb{R}^P$ corresponds to training image patch i and class j. Right: the dictionary elements of $\boldsymbol{D}$ and $\boldsymbol{W}$ are patches.

The discriminative dictionary learning problem (2) is an extension of the basic dictionary learning problem (1), which is obvious from a simple rewriting:

$$\min_{\boldsymbol{D},\boldsymbol{W},\boldsymbol{H}} \left\| \begin{bmatrix} \boldsymbol{Y} \\ \lambda \boldsymbol{Z} \end{bmatrix} - \begin{bmatrix} \boldsymbol{D} \\ \lambda \boldsymbol{W} \end{bmatrix} \boldsymbol{H} \right\|_\mathrm{F}^2 + \gamma \|\mathtt{vec}(\boldsymbol{H})\|_1 . \tag{4}$$

Standard dictionary learning approaches can also be used for this problem. We use the K-SVD algorithm [22] as suggested by Zhang and Li [16], using the efficient implementation from [23] that avoids forming the matrices explicitly. When this algorithm is used to solve (4) we must re-normalize the results to ensure that both $\boldsymbol{D}$ and $\boldsymbol{W}$ have columns of unit 2-norm. (Alternatively, a constrained nonnegative matrix factorization could be considered, as used by Soltani et al. [11]). In all cases, the computational work in solving (2) is larger than that of solving the reconstruction problem; but the dictionary learning is only needed once for a given class of problems.

B. Simultaneous Reconstruction and Segmentation

Our reconstruction model uses the following concepts and techniques, as illustrated in Figure 2.

a) Tomographic Image Reconstruction: Let $\boldsymbol{x} \in \mathbb{R}^n$ represent the unknown attenuation coefficients of the object, which has been discretized into a square domain of $\sqrt{n} \times \sqrt{n}$ pixels. The data vector $\boldsymbol{b}$ represents the attenuation of the X-rays that penetrate the object at n_p different projection angles and recorded in n_d detector pixels, hence $\boldsymbol{b}$ has $m = n_\mathrm{p} n_\mathrm{d}$ elements. The relation between image pixels and data is described by the sparse system matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$. The "naive" tomographic reconstruction problem is the process of computing $\boldsymbol{x}$ given $\boldsymbol{b}$ by solving the linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$.

b) Segmentation: Classification of each pixel x_j assigns this pixel to one of the N_C classes, and we do this via pixel-wise class probabilities. Let $\boldsymbol{\Delta} = [\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \ldots, \boldsymbol{\delta}_{N_\mathrm{C}}] \in \mathbb{R}^{n \times N_\mathrm{C}}$

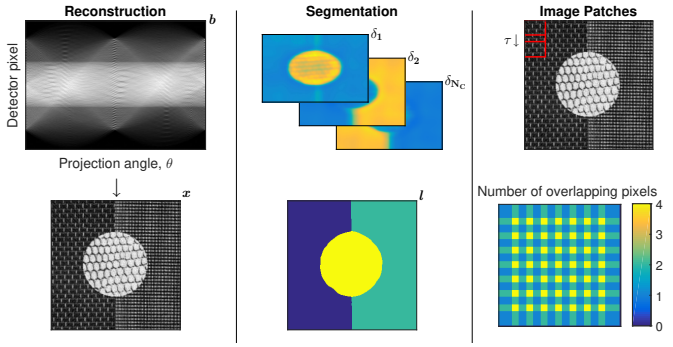

Fig. 2: Illustration of the concepts involved in the proposed algorithm. Left: tomographic image reconstruction from projection data. Middle: segmentation based on pixel-wise class probabilities. Right: division of the image domain into overlapping patches.

be a matrix where each element δ_{jk} represents the probability of pixel x_j belonging to class k, with the constraints

$$\sum_{k=1}^{N_\mathrm{C}} \delta_{jk} = 1\ , \qquad \delta_{jk} \geq 0. \tag{5}$$

Given $\boldsymbol{\Delta}$, the segmentation $\boldsymbol{l} \in \mathbb{N}^n$ is computed by evaluating which of the classes that has the greatest probability for each pixel:

$$l_j = \arg\max_k \left(\delta_{jk}\right), \quad j = 1, 2, \ldots, n. \tag{6}$$

c) Working with Overlapping Image Patches: As the dictionary elements represent small $p \times p$ image patches, a terminology for dividing the image into such patches is required. Let $\boldsymbol{E}_i \in \mathbb{R}^{P \times n}$ be a binary matrix, such that $\boldsymbol{E}_i \boldsymbol{x}$ extracts patch i of the image. The first extractor $\boldsymbol{E}_1$ picks the top left corner of the image. The location of the neighboring patches is controlled with the stride parameter τ, i.e., the number of pixels shifted horizontally or vertically; if $\tau = p$ there is no overlap between the patches. The total number of image patches (and hence the number of extractors) is denoted by N_P.

Given the learned discriminative dictionary $\{\boldsymbol{D}, \boldsymbol{W}\}$ from Section II-A, we can now introduce the model for simultaneous reconstruction and segmentation:

$$\begin{aligned}\min_{\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\Delta}}\ & \lambda_\mathrm{data}^2 \left\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\right\|_2^2 + \lambda_\mathrm{class} \sum_{k=1}^{N_\mathrm{C}} \mathcal{R}_\mathrm{TV}(\boldsymbol{\delta}_k) - \left\|\boldsymbol{\Delta}\right\|_\mathrm{F}^2 \ + \\ & \sum_{i=1}^{N_\mathrm{P}} \left(\left\|\boldsymbol{E}_i\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_i\right\|_2^2 + \sum_{k=1}^{N_\mathrm{C}} \left\|\boldsymbol{E}_i\boldsymbol{\delta}_k - \boldsymbol{W}_k\boldsymbol{\alpha}_i\right\|_2^2 \right) \\ & + \ \gamma_\mathrm{sc} \left\|\boldsymbol{\alpha}\right\|_1 ,\end{aligned} \tag{7}$$

subject to the constraints in (5). Here, $\boldsymbol{\delta}_k$ are the columns of the matrix $\boldsymbol{\Delta}$, the vectors $\boldsymbol{\alpha}_i \in \mathbb{R}^{N_\mathrm{D}}$ are the sparse coding coefficients for image patch i, the vector $\boldsymbol{\alpha} \in \mathbb{R}^{N_\mathrm{P} N_\mathrm{D}}$ is obtained by stacking these vectors, and $\|\boldsymbol{\alpha}\|_1 = \sum_{i=1}^{N_P} \|\boldsymbol{\alpha}_i\|_1$. We incorporate the term $-\|\boldsymbol{\Delta}\|_\mathrm{F}^2$ in order to enforce sparsity in the computed segmentation [24].

When using a forward finite-difference approximation, the Total Variation (TV) function is given by

$$\mathcal{R}_\mathrm{TV}(\boldsymbol{\delta}_k) = \sum_{j \in \mathcal{J}} \sqrt{(\delta_{jk} - \delta_{j'k})^2 + (\delta_{jk} - \delta_{j''k})^2 + \epsilon^2}, \tag{8}$$

where $\mathcal{J}$ represent the pixel indices of the image domain, while j' and j'' denote neighbor pixels in the horizontal and vertical directions, respectively. In order to make the function differentiable at all points, a small constant $\epsilon = 10^{-3}$ is added.

The first term in the model (7) is a least-squares data fitting term. The next two terms enforce regularization on the class probabilities $\boldsymbol{\Delta}$: the TV of $\boldsymbol{\delta}_k$ ensures that neighboring pixels should have similar labelling, i.e., piecewise constant class probabilities; the Frobenius norm forces $\boldsymbol{\Delta}$ towards a state where each pixel has a probability of 1 for a particular class. The last terms (the sum over image patches and the 1-norm) enforce joint sparse representation of the reconstruction and the class probabilities using the discriminative dictionaries.

The model contains the following parameters:

- λ_data represents the amount of trust in the measured data.
- λ_class controls the amount of TV regularization.
- γ_sc controls the degree of sparsity.

III. The Optimization Algorithm

The nonconvex minimization problem in Eq. (7) is solved by a standard iterative algorithm which alternates between three separate subproblems, where one variable is updated and the other two are fixed.

Subproblem 1: The subproblem for computing the sparse coding $\boldsymbol{\alpha}$ has the form

$$\begin{aligned}\min_{\boldsymbol{\alpha}} & \sum_{i=1}^{N_\mathrm{P}} \left(\left\|\boldsymbol{E}_i\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_i\right\|_2^2 + \sum_{k=1}^{N_\mathrm{C}} \left\|\boldsymbol{E}_i\boldsymbol{\delta}_k - \boldsymbol{W}_k\boldsymbol{\alpha}_i\right\|_2^2 \right) \\ & + \ \gamma_\mathrm{sc} \left\|\boldsymbol{\alpha}\right\|_1 .\end{aligned} \tag{9}$$

The combination of a least squares term with 1-norm regularization can be solved efficiently with proximal forward-backward algorithms, and we use the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [25]. Each extracted patch is treated independently, which provides a potential for parallelization in the implementation.

Subproblem 2: The subproblem for computing the reconstruction $\boldsymbol{x}$ has the form

$$\min_{\boldsymbol{x}} \lambda_\mathrm{data}^2 \left\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\right\|_2^2 + \sum_{i=1}^{N_\mathrm{P}} \left\|\boldsymbol{E}_i\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_i\right\|_2^2 . \tag{10}$$

A bit of rewriting displays the least squares nature of this problem:

$$\min_{\boldsymbol{x}} \left\| \begin{bmatrix} \lambda_\mathrm{data}\boldsymbol{A} \\ \boldsymbol{E}_1 \\ \boldsymbol{E}_2 \\ \vdots \\ \boldsymbol{E}_{N_\mathrm{P}} \end{bmatrix} \boldsymbol{x} - \begin{bmatrix} \lambda_\mathrm{data}\boldsymbol{b} \\ \boldsymbol{D}\boldsymbol{\alpha}_1 \\ \boldsymbol{D}\boldsymbol{\alpha}_2 \\ \vdots \\ \boldsymbol{D}\boldsymbol{\alpha}_{N_\mathrm{P}} \end{bmatrix} \right\|_2^2 . \tag{11}$$

This can be efficiently solved with the CGLS algorithm [26], which does not require the stacked coefficient matrix to be explicitly formed.

Subproblem 3: The subproblem for computing the class probabilities $\boldsymbol{\Delta}$ has the form

$$\begin{aligned}\min_{\boldsymbol{\Delta}} \lambda_{\text{class}} \sum_{k=1}^{N_\text{C}} \mathcal{R}_\text{TV}(\boldsymbol{\delta}_k) - \|\boldsymbol{\Delta}\|_\text{F}^2 \\ + \sum_{i=1}^{N_\text{P}} \sum_{k=1}^{N_\text{C}} \|\boldsymbol{E}_i\boldsymbol{\delta}_k - \boldsymbol{W}_k\boldsymbol{\alpha}_i\|_2^2\end{aligned} \tag{12}$$

with the constraints in (5). This problem is solved with the Frank-Wolfe algorithm [27].

An initialization of two of the three variables is required. It is difficult to assign an initial sparse coding, and we therefore need to initialise $\boldsymbol{x}$ and $\boldsymbol{\Delta}$.

Initial Reconstruction: $\boldsymbol{x}^\text{init}$ can be estimated with any classic reconstruction technique; preferably a fast method that requires few parameters to tune and does not amplify noise. In this work, we use Tikhonov regularization which is implemented using CGLS:

$$\boldsymbol{x}^\text{init} = \arg\min_{\boldsymbol{x}} \lambda_\text{data}^2 \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \lambda_\text{Tik}^2 \|\nabla\boldsymbol{x}\|_2^2 . \tag{13}$$

Here, we use the short-hand notation $\nabla\boldsymbol{x}$ to denote a vector of length n whose jth element is the gradient magnitude $\sqrt{(x_j - x_{j'})^2 + (x_j - x_{j''})^2}$, using the notation from (8).

Initial Class Probability: We first estimate a sparse representation of the initial reconstruction using the dictionary $\boldsymbol{D}$:

$$\boldsymbol{\alpha}^\text{init} = \arg\min_{\boldsymbol{\alpha}} \sum_{i=1}^{N_\text{P}} \left\|\boldsymbol{E}_i\boldsymbol{x}^\text{init} - \boldsymbol{D}\boldsymbol{\alpha}_i\right\|_2^2 + \gamma_\text{sc} \|\boldsymbol{\alpha}\|_1 . \tag{14}$$

This is a simplified version of (9) and it is solved in the same manner. Note that this result is not considered as an initialization of the sparse coding, since the sparse representation is only based on the intensity image, and not on the class probability (which is not estimated yet).

In order to initialize $\boldsymbol{\Delta}$, sparse coding is then directly applied with $\boldsymbol{W}$, similarly to what we did in Subproblem 3, and solved in the same way:

$$\begin{aligned}\boldsymbol{\Delta}^\text{init} = \arg\min_{\boldsymbol{\Delta}} \sum_{i=1}^{N_\text{P}} \sum_{k=1}^{N_\text{C}} \left\|\boldsymbol{E}_i\boldsymbol{\delta}_k - \boldsymbol{W}_k\boldsymbol{\alpha}_i^\text{init}\right\|_2^2 \\ + \lambda_\text{class} \sum_{k=1}^{N_\text{C}} \mathcal{R}_\text{TV}(\boldsymbol{\delta}_k),\end{aligned} \tag{15}$$

with the constraint in (5).

IV. Numerical Experiments

The focus of these experiments is primarily on exploring and understanding the tomographic reconstruction and segmentation abilities of the proposed method. Hence, the dictionary learning problem is downplayed here.

For the experiments, we generate phantom images with textures taken from the Brodatz database [28]. Textured images are interesting to study, as they represent problems where it is difficult to manually select and formulate the appropriate features to describe the prior information. With these texture phantoms we intend to study

- how our method compares against similar methods, and
- how our method generalizes, i.e., how it behaves when applied to data beyond what was seen in the training set.

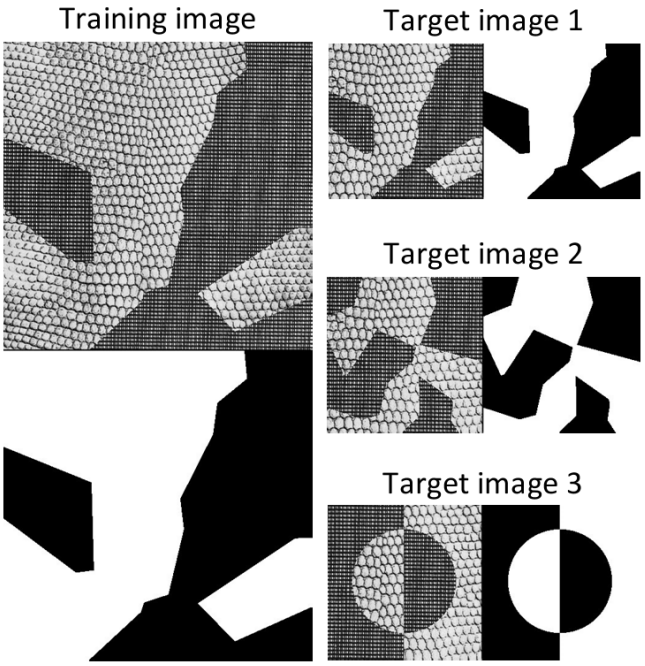

Fig. 3: The textured phantoms used in our experiments. Due to the close similarity between the training data and target image 1, this particular case represents a scenario where the learned prior is close to being ideal. Target images 2 and 3 represent scenarios where the learned prior is incomplete.

The phantoms used in our numerical experiments are shown in Figure 3. We first generate a large training image from which we extract randomly placed patches to compute the dictionaries. For the reconstruction experiments we generate three smaller target phantoms, all of them with similar textures.

Target image 1 has class interface placements that are very similar to the training image, only with small variations. This case can therefore be considered as "inverse crime." With this test problem we study how our method behaves when the class interfaces are well represented in the dictionary.

Target images 2 and 3 have different class interface comparing with the training image, such that the images are not completely identical. In these cases, we test how well our method handles phantoms containing class interfaces that are not part of the training data. In a real scenario, one should ideally have enough training data to learn all possible combinations and orientations of the class interfaces.

A. Four Competing Methods

The proposed method provides both a reconstruction and segmentation of the object, and so should the methods we want to compare it against. We choose one of the latest SRS methods, from [8], as method 1; most other techniques provide only a solution to one of the two unknowns. While we, in principle, could combine separate reconstruction and segmentation methods, many of them are not suited for textured images

which would result in an unfair comparison. For the remaining competing methods we therefore use dictionary-based methods that have access to the same prior information as ours.

In methods 2 and 3 we use Discriminative Dictionary Segmentation (DDS), a stand-alone segmentation method based on a known discriminative dictionary $\{\boldsymbol{D}, \boldsymbol{W}\}$ that can be applied to any reconstruction. It is a modified version of the segmentation method proposed in [20]. The reconstructed image $\boldsymbol{x}$ is first split into overlapping patches, and the sparse representation of each patch is found in the dictionary $\boldsymbol{D}$ by solving (14). Then we use the resulting $\boldsymbol{\alpha}$ with the discriminative part $\boldsymbol{W}$ of the dictionary. For each class k the "evidence" from the estimated sparse codes is accumulated:

$$\boldsymbol{v}_k = \sum_{i=1}^{N_\mathrm{T}} \boldsymbol{E}_i^T \boldsymbol{W}_k \boldsymbol{\alpha}_i, \qquad k = 1, 2, \ldots, N_\mathrm{C}. \tag{16}$$

The segmentation is then obtained by assigning the kth label with the most evidence, similar to (6).

Method 1: This method is from [8] and it represents a non-dictionary based approach. It is a simplified version of the algorithm from [6], and the required prior information is the *mean attenuation coefficient* for each of the classes. The underlying assumption is that the relationship between the reconstructed intensities and the class probabilities can be modelled using Gaussian distributions with joint variance. See [8] for details.

Method 2: This method uses a basic reconstruction technique followed by a dictionary-based post-processing segmentation.

- Reconstruction: Tikhonov regularization solved with CGLS, cf. (13).
- Segmentation: DDS as presented above.

Method 3: This method uses the intensity dictionary $\boldsymbol{D}$ to make an improved reconstruction from the projection data, and then it performs a post-processing segmentation of the reconstruction.

- Reconstruction: uses the method proposed by Xu et al. [10] with slight modifications,

$$\min_{\boldsymbol{x},\boldsymbol{\alpha}} \lambda_\mathrm{data} \left\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\right\|_2^2 + \sum_{i=1}^{N_\mathrm{T}} \left(\left\|\boldsymbol{E}_i\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_i\right\|_2^2 \right) + \gamma_\mathrm{sc} \left\|\boldsymbol{\alpha}\right\|_1 . \tag{17}$$

  The initial reconstruction $\boldsymbol{x}^\mathrm{init}$ is the Tikhonov solution from method 1. The optimization is done with the same algorithms as presented for subproblems 1 and 2, cf. Section III.
- Segmentation: DDS as presented above. Note that $\boldsymbol{\alpha}$ is found as a part of the reconstruction stage, and the segmentation is done by directly applying (16).

Method 4: This is our new method, where segmentation and reconstruction and performed as a part of a combined model.

Methods 2–4 are increasingly complex, seeking to harness more of the available prior information presented by discriminative dictionary. Method 2 uses the prior only to perform a stand-alone segmentation. Method 3 uses part of the prior information to improve the reconstruction before performing a stand-alone segmentation. Finally, method 4 which uses the information in a combined reconstruction and segmentation model.

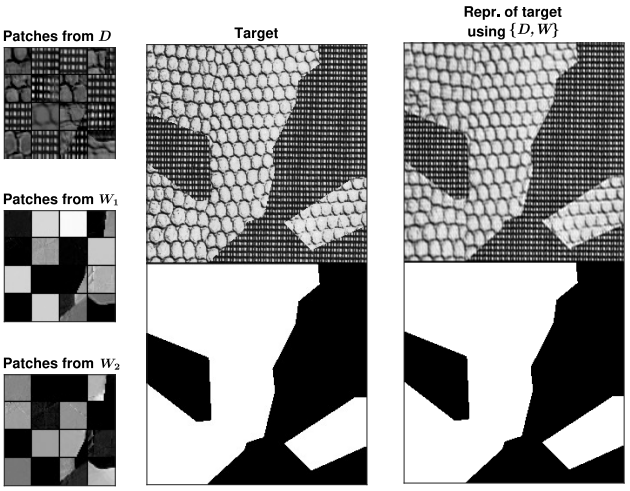

Fig. 4: Left to right: small subsets of the dictionary elements of $\boldsymbol{D}$ and $\boldsymbol{W}$, target image 1, and the representation of this image using $\{\boldsymbol{D}, \boldsymbol{W}\}$.

B. Discriminative Dictionary Learning

The first stage of the proposed method is the training of the dictionary. The following values for the parameters of the learning problem were set manually, such that we obtain a qualitatively good representation of the target image and its segmentation.

- Patch size: p = 23.
- Number of training patches: N_T = 30,000.
- Dictionary size: N_D = 1225.
- Learning sparsity weight: γ = 10.
- Discriminative weighting: λ^2 = 0.5.

Figure 4 illustrates a subset of the learned dictionary elements and the corresponding representation, using the dictionary, of target image 1. While it is qualitatively good, it is not perfect due to the finite size of the dictionary; e.g., it has 2 pixel misclassifications.

C. SRS Experiments

A 2D parallel-beam CT scenario is simulated using the function `paralleltomo` from the AIR TOOLS package [29]. Each phantom is generated in a square domain of 254×254 pixels, i.e., there are $n = 254^2 = 64,516$ unknowns.

The detector has full coverage of the object at any projection angle, the number of detectors is $n_\mathrm{d} = 360$, and we use a constant angular spacing of the rays in the interval of $[0°, 180°]$, using either n_p = 180, 90, 60 and 30 projection angles. These settings correspond to scenarios with even- and under-determined problems of m/n = 1.00, 0.50, 0.33 and 0.17, respectively.

TABLE I: Settings of the parameters in Methods 2–4 for our experiments.

Projections	λ^2_{data}	λ^2_{class}	γ_{sc}	τ	λ^2_{Tik}
180	$7 \cdot 10^{-5}$	0.3	0.175	7	205
90	$9 \cdot 10^{-5}$	0.4	0.150	7	205
60	$9 \cdot 10^{-5}$	0.5	0.125	7	205
30	$9 \cdot 10^{-5}$	0.6	0.125	7	205

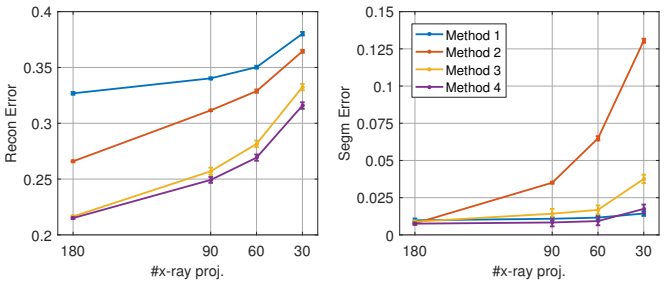

Fig. 5: Reconstruction and segmentation errors (18) for the four methods as a function of the number n_{p} of projection angles. The results are averaged over 7 random realizations of the noise, and we also show ± 2 standard deviations on either side.

The simulated projection data is given by $\boldsymbol{b} = \boldsymbol{A}\,\boldsymbol{x}^{\text{GT}} + \boldsymbol{e}$, where $\boldsymbol{x}^{\text{GT}}$ denotes the ground truth (the true object) and $\boldsymbol{e}$ is a vector of Gaussian noise scaled such that $\|\boldsymbol{e}\|_2 / \|\boldsymbol{A}\,\boldsymbol{x}^{\text{GT}}\|_2 = 0.05$,

The parameters for the different reconstruction methods are set manually by tuning them to yield approximately optimal results. For Method 1 we use $\lambda_{\text{data}} = 0.075$, $\lambda_{\text{class}} = 2.0$, $\mu_1 = 0.1$ and $\mu_2 = 0.85$ (see [8] for explanations). For Methods 2–4 we use the parameters listed in Table I, and we use the same settings when applicable to make the results comparable.

Given the ground truth image $\boldsymbol{x}^{\text{GT}}$ and the corresponding segmentation $\boldsymbol{l}^{\text{GT}}$, we define error measures for the computed reconstruction $\boldsymbol{x}$ and segmentation $\boldsymbol{l}$ as

$$\frac{\|\boldsymbol{x}^{\text{GT}} - \boldsymbol{x}\|_2^2}{\|\boldsymbol{x}^{\text{GT}}\|_2^2} \qquad \text{and} \qquad \frac{1}{N}\sum_{j=1}^{N} \mathcal{I}(\boldsymbol{l}_j^{\text{GT}} \neq \boldsymbol{l}_j), \tag{18}$$

where $\mathcal{I}$ is a logical indicator function.

V. Results and Discussion

A. Comparison of the Methods

Each of the four methods are applied to the same problems with 7 random noise realizations, and with 4 different numbers of projection angles. The resulting reconstruction and segmentation errors of these 28 experiments are summarized in Figure 5. The qualitative difference between the methods is illustrated in Figure 6 which shows the results for one particular noise realization using n_{p} = 180 and 30.

Re. the reconstructions: Methods 1 lacks sharpness in the reconstructed image even with $n_{\text{p}} = 180$ projections, which is expected since Tikhonov regularization is unsuited for the high frequency textures in the phantom. Methods 2 and 3 are able to use the prior information to improve the reconstruction significantly. The reconstructions from methods 3 and 4 are equally good with sufficient projection data; however, with fewer projections the advantage of our method becomes increasingly evident. The issue with method 3 is that its choice of dictionary elements does not utilize information about the classes, and hence it tends to fit the reconstruction to the noise. Method 4 (our method), on the other hand, takes advantage of the segmentation to restrict the choice of dictionary elements to also match the class probability.

Re. the segmentations: With enough projections, all four methods are able to provide good segmentations. Method 1 provides surprisingly good segmentations, due to the simple structure of the phantom (but note that the method lacks sharpness at the class interfaces). Part of segmentation errors that are present for method 2 are corrected with method 3, because the sparse coding is obtained together with the reconstruction in method 3, which avoids the error propagation from the reconstruction. The small and spurious misclassifications that are seen with method 3 are handled nicely by the TV in method 4 (our method), in particular when the amount of data is reduced.

B. Limitations of the Prior

In this experiment we consider the proposed method for some cases that challenges the learned prior. Difficulties can arise from having less projection data, or from using data containing features that were not available from the dictionary.

Projection data for target images 2 and 3 (cf. Figure 3) were simulated using $n_{\text{p}} = 90$ projection angles. The reconstruction results are shown in Figure 7, along with an additional reconstruction example using target image 1 with only 30 projections and a different noise realization from the one in Figure 6. We see that in the segmentation sharp and narrow class interfaces tend to be smoothed, while curved interfaces tend to be more straight. The reason is that curved or sharp interfaces are not represented in the training image as well as the dictionary, see the training image shown in Figure 3. Therefore, it is very important to choose training data that preferably include all possible combinations and orientations of the class interfaces. Furthermore, the higher accuracy the learnt dictionary is, the better performance our joint method will provide. In addition, the third example illustrates what happens in the reconstruction when regions of the segmentation are incorrect, e.g., the top-right corner.

C. Multi-Class Phantoms

In this experiment we apply the proposed method to a phantom with 4 classes, which is shown in the left part of Figure 8. The discriminative dictionary was trained using a large image that includes all class interfaces in the phantom. In method 1 we used $\mu_1 = 0.2$, $\mu_2 = 0.4$, $\mu_3 = 0.45$ and $\mu_4 = 0.3$, and for method 2–4 we use the same parameter setting. We computed the reconstruction and segmentation of the target phantom from the data with 90 projection angles and 5% Gaussian noise, the results are shown and compared to the ground truth in Figure 8. It is obvious that method

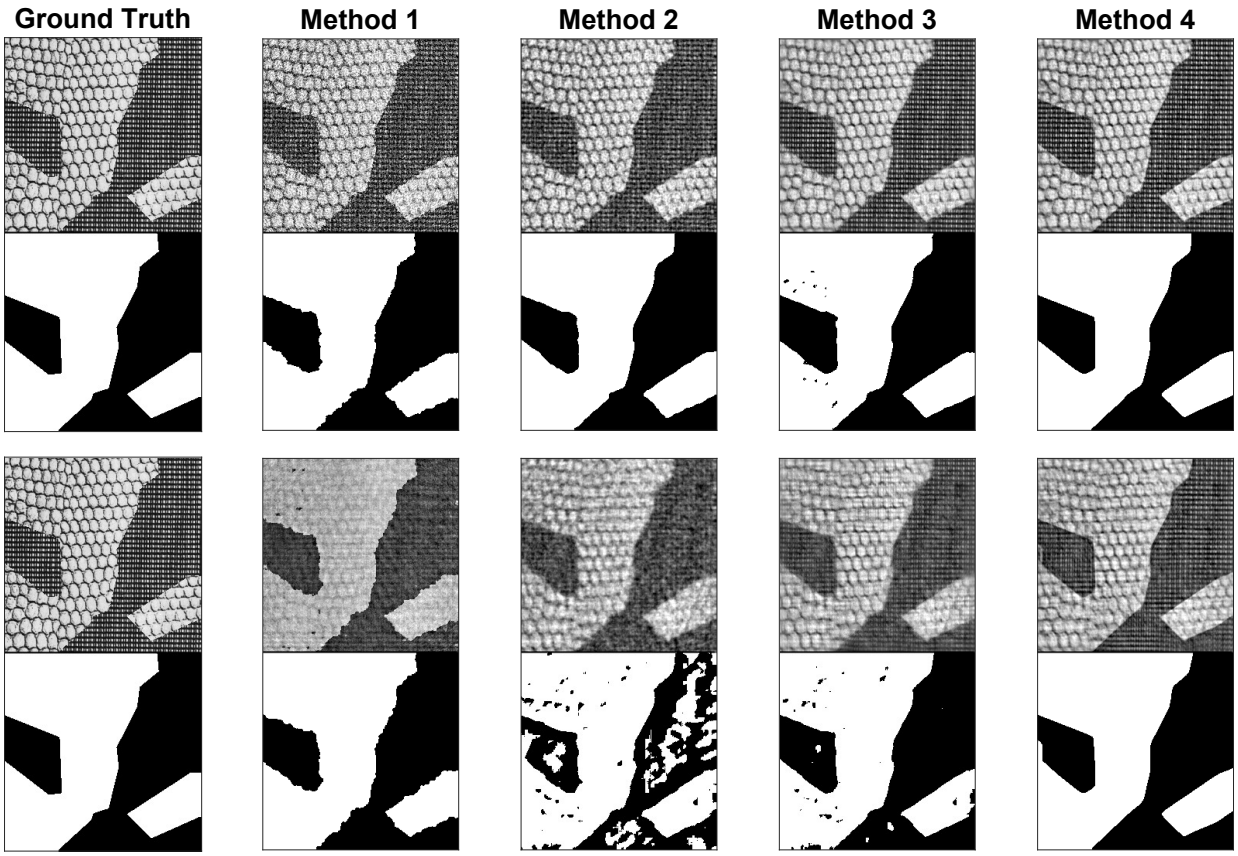

Fig. 6: Visual comparison of the results from the four methods with $n_{\mathrm{p}} = 180$ (top) and $n_{\mathrm{p}} = 30$ (bottom).

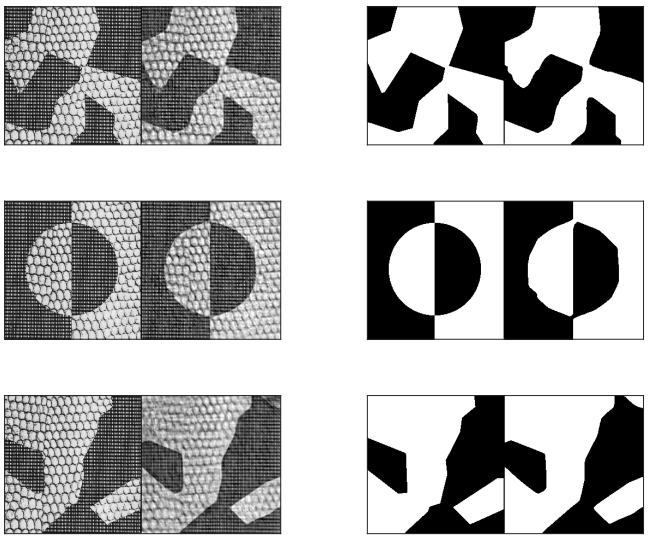

Fig. 7: Reconstructions and segmentations of the three phantoms (top to bottom) using 90, 90 and 30 projections, respectively. The ground truth image and corresponding segmentation are shown in the left parts of the images, while the results from method 4 are shown in the right parts.

3 and 4 provide the best results, and quantitatively they are equally good. Comparing the segmentation, we can see that with method 4 the class interfaces are more straight, which is due to the TV regularization.

D. Test on Fibrous Samples

In the final experiment we study the performance when our method is applied to real fibrous samples. We simulated 2D X-ray data with inspiration from cross-sections of fibrous samples. These structures could represent nerves or muscles in biological tissues, or pipe- or cable-like structures in material samples. The challenge in this experiment is to differentiate between the fiber interior and the background, both of which have very similar intensity and variation. The test images and results are shown in Figure 9.

The size of the fibers in the test images is approximately 5–25 pixels in diameter. The setting in the discriminative dictionary learning are: patch size $p = 17$, number of training patches $N_{\mathrm{T}} = 60,000$, dictionary size $N_{\mathrm{D}} = 1225$, learning sparsity weight $\gamma = 3$, and discriminative weighting $\lambda^2 = 3$. In the SRS experiment we use $n_{\mathrm{p}} = 90$ projection angles, and the Gaussian noise in the sinogram is scaled as $\|\boldsymbol{e}\|_2/\|\boldsymbol{A}\,\boldsymbol{x}^{\mathrm{GT}}\|_2 = 0.025$. The parameters in method 1 are $\lambda_{\mathrm{data}} = 0.3$, $\lambda_{\mathrm{class}} = 0.8$, $\mu_1 = 0.6$, $\mu_2 = 0.27$, and $\mu_3 = 0.6$. In methods 2–4 we use $\lambda^2_{\mathrm{data}} = 10^{-4}$, $\lambda^2_{\mathrm{class}} = 0.175$, $\gamma_{\mathrm{sc}} = 0.01$, $\tau = 3$, and $\lambda^2_{\mathrm{Tik}} = 20$. All parameters are tuned manually and provide the optimal visual results.

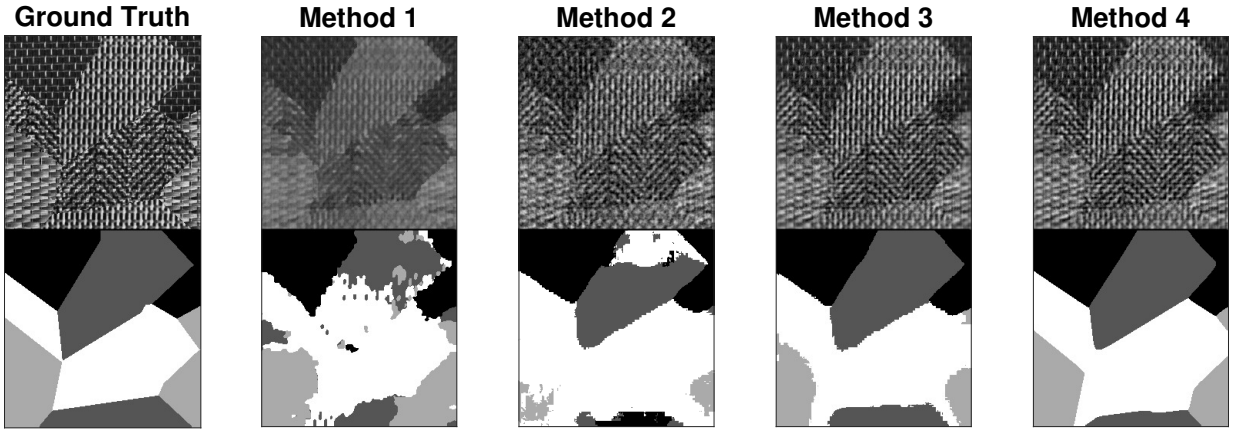

Fig. 8: Comparison of method 1–4 applied to a 4-class phantom with $n_{\mathrm{p}} = 90$ projection angels. The values of (reconstruction error, segmentation error) for the four methods are (0.42, 0.22), (0.35, 0.30), (0.28, 0.12) and (0.28, 0.12), respectively.

Comparing the reconstruction results, we see that there are less artefacts from method 4 (our method), and its background is smoother. For segmentation, method 1 cannot distinguish the interior of the fibers from the background, since incorporating a prior on image smoothness by adding a Tikhonov regularization term is unsuited for segmenting classes with similar intensity. Compared with methods 2 and 3, method 4 is able to correctly capture more interior of the fibers. Method 4 also outperforms the other methods quantitatively according to the reconstruction error and segmentation error.

VI. Conclusion

In this paper, we propose a new simultaneous reconstruction and segmentation (SRS) model incorporating a discriminative dictionary for computed tomography. In our SRS formulation, through a joint sparse coding the segmentation acts as a regularizer and brings more prior knowledge for the reconstruction. In addition, because of using the discriminative dictionary, our method is able to segment different textures, which are usually difficult to be formulated in closed form. In order to deal with dictionary more efficiently, in the future we intend to introduce more advanced techniques, e.g. nonnegative matrix factorization or tensor dictionary, into our method.

Acknowledgment

This work was supported by the European Research Council under Grant No. 291405 (HD-Tomo) and Grant 11701388 from the National Natural Science Foundation of China.

References

[1] P. C. Hansen, *Discrete Inverse Problems: Insight and Algorithms*. SIAM, PA, 2010.

[2] S. Yoon, A. R. Pineda, and R. Fahrig, "Simultaneous segmentation and reconstruction: A level set method approach for limited view computed tomography," *Medical Physics*, vol. 37, no. 5, pp. 2329–2340, 2010.

[3] K. Batenburg, J. Sijbers, H. Poulsen, and E. Knudsen, "DART: a robust algorithm for fast reconstruction of three-dimensional grain maps," *Journal of Applied Crystallography*, 2010.

[4] R. Ramlau and W. Ring, "A Mumford–Shah level-set approach for the inversion and segmentation of X-ray tomography data," *Journal of Computational Physics*, vol. 221, no. 2, pp. 539–557, 2007.

[5] D. Van de Sompel and M. Brady, "Simultaneous reconstruction and segmentation algorithm for positron emission tomography and transmission tomography," in *Proc. 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro.* IEEE, 2008, pp. 1035–1038.

[6] M. Romanov, A. B. Dahl, Y. Dong, and P. C. Hansen, "Simultaneous tomographic reconstruction and segmentation with class priors," *Inverse Problems in Science and Engineering*, vol. 24, no. 8, pp. 1432–1453, 2016.

[7] F. Lauze, Y. Quéau, and E. Plenge, "Simultaneous reconstruction and segmentation of CT scans with shadowed data," in *Scale Space and Variational Methods in Computer Vision: 6th International Conference*, F. Lauze, Y. Dong, and A. B. Dahl, Eds. Springer, 2017, pp. 308–319.

[8] H. M. Kjer, Y. Dong, and P. C. Hansen, "User-friendly simultaneous tomographic reconstruction and segmentation with class priors," in *Scale Space and Variational Methods in Computer Vision: 6th International Conference*, F. Lauze, Y. Dong, and A. B. Dahl, Eds. Springer, 2017, pp. 260–270.

[9] D. Karimi and R. K. Ward, "Patch-based models and algorithms for image processing: a review of the basic principles and methods, and their application in computed tomography," *International Journal of Computer Assisted Radiology and Surgery*, vol. 11, no. 10, pp. 1765–1777, 2016.

[10] Q. Xu, H. Yu, X. Mou, L. Zhang, J. Hsieh, and G. Wang, "Low-dose X-ray CT reconstruction via dictionary learning," *IEEE Transactions on Medical Imaging*, vol. 31, no. 9, pp. 1682–1697, 2012.

[11] S. Soltani, M. S. Andersen, and P. C. Hansen, "Tomographic image reconstruction using training images," *Journal of Computational and Applied Mathematics*, vol. 313, pp. 243–258, 2017.

[12] S. Soltani, M. E. Kilmer, and P. C. Hansen, "A tensor-based dictionary learning approach to tomographic image reconstruction," *BIT Numerical Mathematics*, vol. 56, pp. 447–454, 2016.

[13] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 948–958, 2010.

[14] I. Tosic and P. Frossard, "Dictionary learning," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 27–38, 2011.

[15] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *2010 Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 3501–3508.

[16] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *2010 Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 2691–2698.

[17] Y. Chen, J. Liu, Y. Hu, J. Yang, L. Shi, H. Shu, Z. Gui, G. Coatrieux, and L. Luo, "Discriminative feature representation: an effective post-processing solution to low dose ct imaging," *Physics in Medicine and Biology*, no. 6, 2017.

[18] Y. Chen, L. Shi, Q. Feng, J. Yang, H. Shu, L. Luo, J.-L. Coatrieux, and W. Chen, "Artifact suppressed dictionary learning for low-dose ct image processing," *IEEE Transaction on Medical Imaging*, no. 12, 2014.

[19] J. Liu, J. Ma, Y. Zhang, Y. Chen, J. Yang, H. Shu, L. Luo, G. Coatrieux, W. Yang, Q. Feng, and W. Chen, "Discriminative feature representation

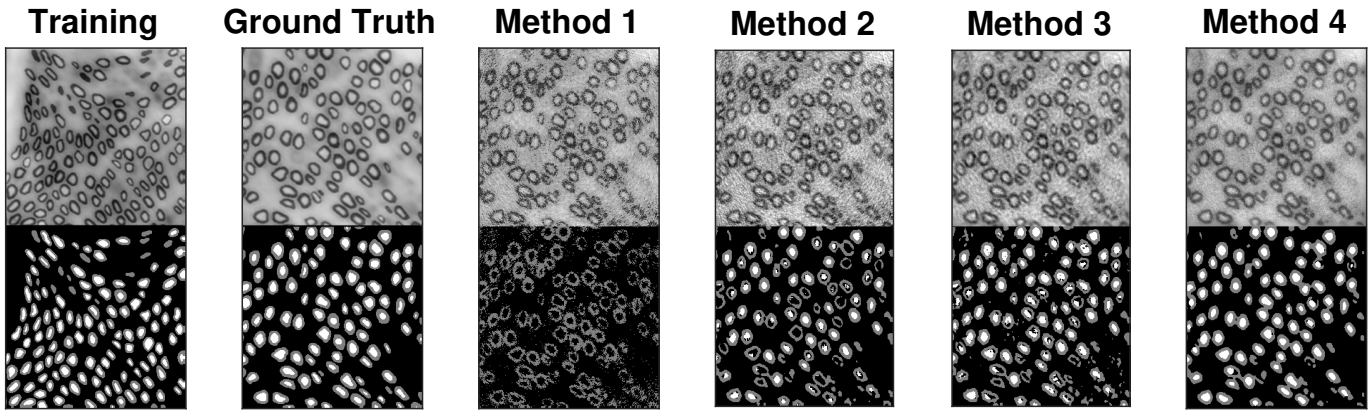

Fig. 9: Experiment with fibrous samples with $n_{\mathrm{p}} = 90$ projection angles. The values of (reconstruction error, segmentation error) for the four methods are (0.18, 0.21), (0.17, 0.11), (0.11, 0.08), and (0.10, 0.07), respectively.

to improve projection data inconsistency for low dose ct imaging," *IEEE Transaction on Medical Imaging*, no. 12, 2017.
[20] T. Tong, R. Wolz, P. Coupé, J. V. Hajnal, D. Rueckert, and The Alzheimer's Disease Neuroimaging Initiative, "Segmentation of MR images via discriminative dictionary learning and sparse coding: Application to hippocampus labeling," *NeuroImage*, vol. 76, pp. 11–23, 2013.
[21] T. M. Buzug, *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*. Springer, 2008.
[22] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
[23] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit," *CS Technion*, vol. 40, no. 8, pp. 1–15, 2008.
[24] P. Li, S. Sundar Rangapuram, and M. Slawski, "Methods for sparse and low-rank recovery under simplex constraints," 2016, preprint on Arxiv at https://arxiv.org/pdf/1605.00507.pdf.
[25] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.
[26] Å. Björck, *Numerical Methods for Least Squares Problems*. SIAM, PA, 1996.
[27] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, Belmont, 1999.
[28] P. Brodatz, *Textures: A Photographic Album for Artists and Designers*. Dover, 1966.
[29] P. C. Hansen and M. Saxild-Hansen, "AIR Tools - A MATLAB package of algebraic iterative reconstruction methods," *Journal of Computational and Applied Mathematics*, vol. 236, no. 8, pp. 2167–2178, 2012.

Yiqiu Dong was born in 1980 in Shandong, China. She received the B.Sc. degree in mathematics from Yantai University, Yantai, China, in 2002 and the Ph.D. degree in mathematics from Peking University, Beijing, China, in 2007.She is currently associate professor in the Technical University of Denmark. Her research area includes mathematical imaging, inverse problem and variational methods, matrix application and computation, and optimization methods.

Per Christian Hansen received his PhD in 1985 and his DrTechn habilitation in 1996, both from the Technical University of Denmark. He is professor of scientific computing, and he works with matrix computations and numerical regularization algorithms. He has published 100 papers in leading journals, and his books on numerical methods for inverse problems are widely used. He has developed several related software packages. He is a SIAM Fellow.

Hans Martin Kjer received his PhD from the Technical University of Denmark in 2015. He is currently a postdoc and works on image processing and analysis from Computed Tomography and Magnetic Resonance Imaging.