Accelerating Convolutional Sparse Coding for Curvilinear Structures Segmentation by Refining SCIRD-TS Filter Banks

Annunziata, Roberto; Trucco, Emanuele

Published in:
IEEE Transactions on Medical Imaging

DOI:
10.1109/TMI.2016.2570123

Publication date:
2016

Document Version
Peer reviewed version

Accelerating Convolutional Sparse Coding for Curvilinear Structures Segmentation by Refining SCIRD-TS Filter Banks

Roberto Annunziata* and Emanuele Trucco

***Abstract*—Deep learning has shown great potential for curvilinear structure (e.g. retinal blood vessels and neurites) segmentation as demonstrated by a recent auto-context regression architecture based on filter banks learned by convolutional sparse coding. However, learning such filter banks is very time-consuming, thus limiting the amount of filters employed and the adaptation to other data sets (i.e. slow re-training). We address this limitation by proposing a novel acceleration strategy to speed-up convolutional sparse coding filter learning for curvilinear structure segmentation. Our approach is based on a novel initialisation strategy (*warm start*), and therefore it is different from recent methods improving the optimisation itself. Our warm-start strategy is based on carefully designed hand-crafted filters (SCIRD-TS), modelling appearance properties of curvilinear structures which are then refined by convolutional sparse coding. Experiments on four diverse data sets, including retinal blood vessels and neurites, suggest that the proposed method reduces significantly the time taken to learn convolutional filter banks (i.e. up to -82%) compared to conventional initialisation strategies. Remarkably, this speed-up does not worsen performance; in fact, filters learned with the proposed strategy often achieve a much lower reconstruction error and match or exceed the segmentation performance of random and DCT-based initialisation, when used as input to a random forest classifier.**

***Index Terms*—convolutional sparse coding, segmentation, retinal blood vessels, neurites.**

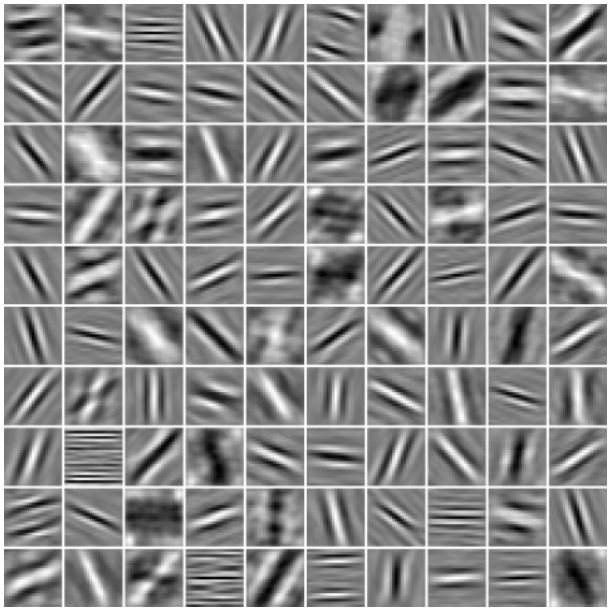

Fig. 1. A filter bank learned using convolutional sparse coding with random initialisation. The DRIVE data set was used for this experiment.

I. Introduction and Related Work

AUTOMATED segmentation of curvilinear structures such as retinal blood vessels and neurites is a particularly active area of research, e.g. [1]–[20]. This task is critical for a whole category of medical image analysis algorithms [21], for instance: (1) screening or monitoring diseases such as diabetic retinopathy, glaucoma, or age-related macula degeneration; (2) computer-assisted diagnosis and risk stratification; (3) biomarkers, i.e., determine whether the occurrence of measurable features in the images is associated with specific conditions [22], [23]. In neuroscience research, reconstructing neuronal trees is a fundamental step to better understand how networks of neurons work. Despite recent advances, neuronal reconstructions are largely obtained manually. Given the resolution currently available, reconstructing neurites of a single cell may take months [24].

Most of the existing methods for automated curvilinear structure segmentation rely on hand-crafted filters (henceforth HCFs) designed to model local geometric properties of ideal *tubular* shapes [1], [4], [7], [13], [23]. However, HCFs often require manual parameter tuning (e.g. width, length and orientation) [1], [4], [7], [13], which does not guarantee optimal performance. Today research is moving towards fully/deep learning architectures (henceforth, DLA) given their excellent results on several challenging tasks [25]–[29]. An attractive property of DLAs is their capability of finding the optimal shape of each adopted filter automatically.

Experimental results show that lower layers of DLAs with convolutional structure (e.g. CNN) tend to learn a subset of filters similar to well-known HCFs (e.g. Gabor filters, see [16], [20], [25], [26], [30], [31]). This is also the case for convolutional sparse coding (henceforth, CSC) as shown in Figure 1. We argue that employing such complex architectures to learn filters similar to HCFs is inefficient; a more efficient approach would be finding an optimal parameter setting (e.g. width, length and orientation) for some HCFs automatically and

This research was supported by the EU Marie Curie ITN REVAMMAD, n^o 316990. *Asterisk indicates corresponding author.*

R. Annunziata and E. Trucco are with the School of Science and Engineering (Computing), University of Dundee, Dundee, DD14HN, UK. E-mail:{r.annunziata, e.trucco}@dundee.ac.uk.

Manuscript received Month day, year; revised Month day, year.

learning *only* appearance characteristics not included in the hand-crafted models. These appearance characteristics could be *data-specific* (due to a disease, for instance) or particular structure configurations (and their variations) difficult to model (e.g. crossings, bifurcations, parallel structures).

Recently, an auto-context framework (multi-layer) based on unsupervised filter learning has been shown to outperform CNN and modifications [12] on curvilinear structure segmentation in the medical domain [17], [32]. The framework proposed in [17], [32] relies on filters learned through CSC [11], [16], but learning them is very time-consuming as reported in [11] (several days to learn 121 filters using MATLAB code and state-of-the-art machines). Therefore, the filter bank learned at the first layer is kept unchanged across the other ones, due to the prohibitive cost of learning layer-specific filter banks [32]. This limitation is particularly relevant for medical imaging applications, where the visual appearance of curvilinear structures may vary significantly and the range of acquisition modalities may lead to different image characteristics in terms of contrast and noise. As a consequence, re-training could be necessary to achieve good performance.

Motivated by the above and inspired by the observation that filters learned by CSC for curvilinear structure segmentation are often similar to well-known HCFs, we propose an efficient approach to learning CSC filters.

Our work differs fundamentally from recent acceleration methods like those reported by Heide et al. [31], Bristow et al. [30], and Bao et al. [33], [34], which rely on efficient mathematical formulations to solve the CSC *optimisation* problem. Such methods typically initialise filters with random values or by a discrete cosine transform (henceforth, DCT). While this solution is general and effective, it does not exploit prior knowledge about the target curvilinear structure and its appearance. One of the first attempts to exploit this information was done by Rigamonti et al. [11], who proposed to combine fast HCFs responses with those of a few filters learned by CSC, hence faster to learn. This solution combines the advantages of HCFs and learned filters, i.e. speed and discriminative power, respectively. However, since CSC filters are learned independently of the HCFs used (i.e. CSC filters are learned directly on original image patches), this approach may lead to redundant filters already included in the HCF bank [15].

The main novelty of our acceleration strategy lies in the integration of curvilinear structure modelling within the CSC learning pipeline, with the aim of leveraging prior information about the target application and reducing the training time (if possible without compromising detection performance). For the modelling part, we propose a new formulation of the recent SCIRD ridge detector [13], [23], denoted as SCIRD-TS, which improves the detection of thin structures. Moreover, we formulate the problem of identifying the optimal set of SCIRD-TS filters to be used to initialise the CSC optimisation as a compression task, and motivate the adoption of a K-means algorithm to perform this task.

CSC filter learning is at the core of state-of-the-art curvilinear structure segmentation pipelines (e.g. [17], [32]), therefore our acceleration strategy, combined with state-of-the-art (and future) fast CSC solvers (e.g. [30], [31], [35]), could potentially contribute to advance the field further (e.g., faster re-training for different data sets and curvilinear structures, the possibility to learn a much larger filter bank which could lead to better segmentation performance, among others).

MATLAB implementation of the whole framework can be found at http://staff.computing.dundee.ac.uk/rannunziata/.

II. Proposed Method

We achieve CSC acceleration by a novel *warm-start initialisation* strategy based on SCIRD-TS (HCF ridge detector). Specifically, the proposed warm-start strategy identifies the optimal set of initial filters from a large amount of HCFs generated by spanning the range of parameters related to the structures of interest. It is worth noting that setting the ranges for HCF parameters is very intuitive, as they represent geometric properties of the target structure and their effects can be checked visually. These filters are then *refined* by using CSC to incorporate specific properties of the structures (e.g. retinal blood vessels, neurites) of a specific data set. Intuitively, the speed-up is achieved by learning only the "properties" which have not been modelled and by refining the ones already modelled (e.g. width or elongation).

An overview of the proposed method is shown in Figure 4.

A. Optimal warm-start strategy

SCIRD for thin structures (SCIRD-TS). Curvilinear structures such as blood vessels and neurites share appearance characteristics which can be easily modelled, rather than learned. In recent years, important efforts have been made in this regard and several HCFs have been proposed (e.g., Frangi [1], Gabor [4], OOF [7]). These methods assume that a curvilinear structure is "locally straight" and well contrasted. However, these assumptions are violated by structures such as blood vessels and neurites, appearing fragmented, showing some level of tortuosity or captured with low signal-to-noise ratio. As a consequence, detection performance may degrade significantly. We addressed these modelling issues in our previous work [13], [23] by proposing a novel ridge detector, SCIRD, which adds curvature and contrast invariance to that of previous HCFs (i.e., scale, rotation and elongation).

In [13], [23] (SCIRD), we model a curvilinear structure with a *curved-support* Gaussian function. Then, the curved ridge detection is obtained by measuring the second directional derivative along the gradient of each curved-support Gaussian. This derivation results in a ridge detector which consists of a ratio of first and second derivatives of the curved-support Gaussian function, thus leading to "0/0" indeterminate form in particular cases, e.g. when the first derivatives vanish. Unfortunately, this compromises the detection of thin structures, as shown qualitatively in Figure 2 (second row).

To address this limitation and avoid indeterminate forms, we modify the derivation of the curved-support ridge detector. Specifically, instead of curving the curvilinear structure model (as done for SCIRD), we first derive a *straight* ridge detector. Then, we apply a non-linear transformation to curve the ridge detector. This new ridge detector is therefore curved as SCIRD is, but when adopting straight filters (i.e. curvature is 0), it

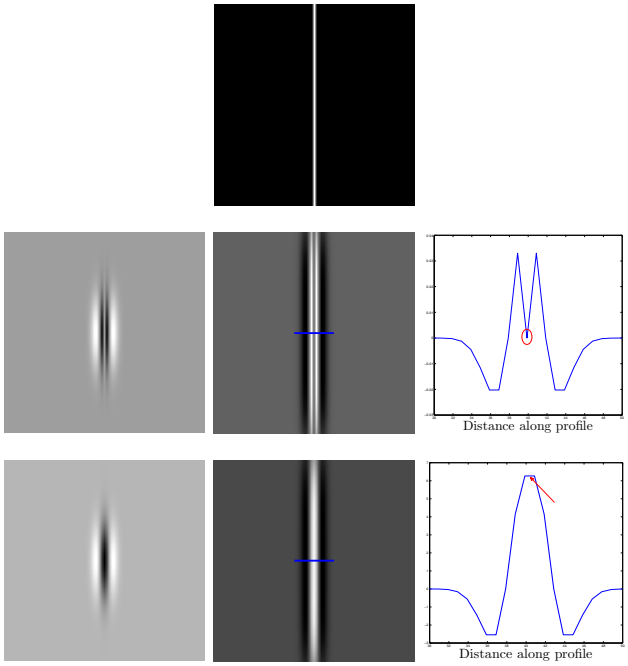

Fig. 2. First row: ideal thin structure (1 pixel wide); second row, from left to right: SCIRD filter, SCIRD response and its cross-sectional profile along the blue line; third row, from left to right: SCIRD-TS filter, SCIRD-TS response and its cross-sectional profile along the blue line. Notice that while the SCIRD response is approximately 0 on the thin structure (i.e. SCIRD does not detect it), the SCIRD-TS one is maximum, hence leading to a correct detection.

does not lead to indeterminate pixel values, as shown Figure 2 (third row). This improves the detection of thin structures, as shown qualitatively in Figure 3 and quantitatively in Figure 7.

Let us model a *straight* ridge-like structure by means of a multivariate zero-mean (n-D) Gaussian function with diagonal covariance matrix,

$$G(\boldsymbol{\varphi};\boldsymbol{\sigma}) = \frac{1}{\sqrt{(2\pi)^n \prod_{i=1}^{n} \sigma_i^2}} \exp\left(-\sum_{i=1}^{n} \frac{\varphi_i^2}{2\sigma_i^2}\right) \quad (1)$$

where $\boldsymbol{\varphi} = (\varphi_1, \varphi_2, \ldots, \varphi_n)$ represents a point in the $\{\varphi\}$ coordinate system, and $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ describes the standard deviation in each direction. A ridge *detector* can be obtained by measuring the contrast between the part inside and outside the ridge [1]. This can be achieved by measuring the second derivative with respect to the variables along which we observe the ridge-like profile. Using the separability property of the n-D Gaussian, one can compute the second derivative with respect to each variable and then combine the results (e.g. by summing up all the contributions). The second derivative of $G(\boldsymbol{\varphi};\boldsymbol{\sigma})$ with respect to the variable φ_j has the form

$$G_{\varphi_j\varphi_j}(\boldsymbol{\varphi};\boldsymbol{\sigma}) = G(\boldsymbol{\varphi};\boldsymbol{\sigma}) \left[\frac{1}{\sigma_j^2}\left(\frac{\varphi_j^2}{\sigma_j^2} - 1\right)\right]. \quad (2)$$

If we assume (without loss of generality) that the structure shows a ridge-like profile only with respect to the coordinate φ_j, the function $G_{\varphi_j\varphi_j}(\boldsymbol{\varphi};\boldsymbol{\sigma})$ represents a ridge detector for *straight* structures. To extend this ridge detector to more general curved-support objects, we consider a non-linear transformation $\mathcal{T} : \mathbb{R}^n \mapsto \mathbb{R}^n$ with $\mathcal{T}(\boldsymbol{x}) = \boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_j, \ldots, \varphi_n)$ of the form

$$\varphi_j = x_j + \sum_{i=1}^{n-1} k_{ji} x_i^2, \qquad 2 \le j \le n \quad (3)$$

and $\varphi_1 = x_1$, where $k_{ji} \in \mathbb{R}$ and x_i are the coordinates of a point in the new $\{x\}$ coordinate system. In the 2-D case (i.e. $n = 2$), applying the transformation $\mathcal{T}$ in Eq. (3) to $G_{\varphi_j\varphi_j}(\boldsymbol{\varphi};\boldsymbol{\sigma})$ in Eq. (2), leads to our SCIRD-TS filter:

$$F(\boldsymbol{x};\boldsymbol{\sigma},k) = \frac{1}{\sigma_2^2\, Z(\boldsymbol{\sigma})}\left[\frac{(x_2 + kx_1^2)^2}{\sigma_2^2} - 1\right] \times \exp\left(-\frac{x_1^2}{2\sigma_1^2}\right) \exp\left(-\frac{\left(x_2 + kx_1^2\right)^2}{2\sigma_2^2}\right), \quad (4)$$

where k_{21} (curvature parameter) is indicated as k for compactness.

To make the ridge detector rotation invariant, SCIRD-TS filters can be simply rotated by θ, applying the rotation matrix to (x_1, x_2). Therefore, we will indicate the SCIRD-TS filters as $F(\boldsymbol{x};\boldsymbol{\sigma},k,\theta)$. For completeness, the unsupervised version of SCIRD-TS can be obtained by taking the maximum response among the ones of all the filters, at each pixel.

A pre-defined convolutional filter bank can be generated by spanning the range of the free parameters σ_1, σ_2, k and θ.

We observed experimentally that our CSC optimisation initialised with random filters tends to converge to "bright" and "dark" filters [1] (for instance, Figure 1 first row - column 4 and 8, respectively). For this reason, for each dark SCIRD-TS filter (e.g. Figure 2, third row - left) we generate its bright counterpart, i.e. $-F(\boldsymbol{x};\boldsymbol{\sigma},k,\theta)$, as well. Moreover, we generate symmetric curved filters, i.e. $0.5F(\boldsymbol{x};\boldsymbol{\sigma},+k,\theta) + 0.5F(\boldsymbol{x};\boldsymbol{\sigma},-k,\theta)$ as they were found to speed-up CSC convergence further.

Let $\mathcal{S} \subseteq \mathbb{R}^p$ the space of all the curvilinear structures in a particular data set, and assume that a subset of them, s, can be detected by using SCIRD-TS filters in the space $\mathcal{F} \subseteq \mathbb{R}^q$ (Figure 4). The parameter ranges of these SCIRD-TS filters can be easily estimated (e.g. by visual inspection of

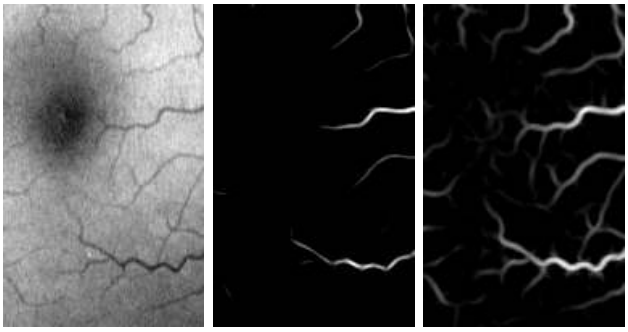

Fig. 3. Detecting thin vessels. Left: original image patch (green channel) showing thin retinal blood vessels around the fovea; middle: enhancement using SCIRD [13], [23]; right: enhancement using the proposed SCIRD-TS. The thin vessels not enhanced by SCIRD are correctly enhanced by SCIRD-TS.

[1]Bright/dark refers to the grey-level of the central pixels

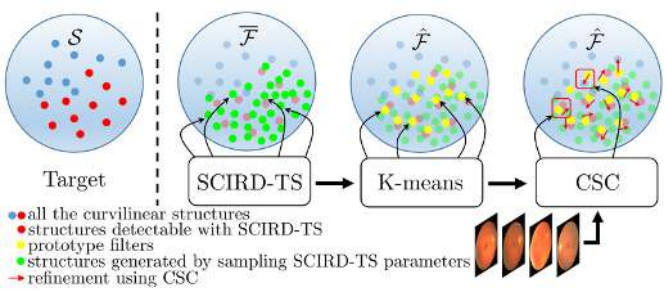

Fig. 4. Block diagram of the proposed method.

the curvilinear structures in $\mathcal{S}$). Uniform sampling of such parameter ranges guarantees a better approximation of the filters in $\mathcal{F}$ as the sampling step $\boldsymbol{\delta}$ vanishes. Let $\overline{\mathcal{F}} \in \mathbb{R}^q$ be the space generated by uniform sampling. So, the first step of our warm-start strategy consists of generating t SCIRD-TS filters in $\overline{\mathcal{F}}$ (t>s, in general) by sampling uniformly and densely its parameter ranges.

Extracting the *optimal* set of prototype filters. Using the entire set of SCIRD-TS filters generated in the previous step is clearly unfeasible (t>$20,000$ with our parameter setting). So, we need to reduce the cardinality of $\overline{\mathcal{F}}$ and map it to a new space $\widehat{\mathcal{F}}$ with a much lower cardinality ($K \ll t$), while still preserving a good representation of $\overline{\mathcal{F}}$ (hence of $\mathcal{F}$). A key requirement for the success of sparse coding dictionary learning strategies is building *incoherent* dictionaries (e.g. [34]). The mutual *incoherence* of a dictionary D can be defined as

$$\mu(D) = \min_{i \neq j} \|\mathbf{d}_i - \mathbf{d}_j\|_2^2, \tag{5}$$

where $\mathbf{d}_i$ and $\mathbf{d}_j$ are two different dictionary elements (or atoms) arranged as one-column vectors. So, a high value of $\mu(D)$ for the learned dictionary is desirable. Moreover, since our overall target is to accelerate CSC, the cardinality reduction should be fast, so that most of the training time is spent on the CSC phase. Of course, sampling uniformly and sparsely SCIRD-TS parameter ranges would be fast, but it would not guarantee high dictionary incoherence.

The compression approach we adopt here to identify the set of K *prototype filters* which represent optimally (in the sense of minimising the quantisation error) the original SCIRD-TS space $\overline{\mathcal{F}}$ is K-means clustering using Euclidean distance[2]. K-means clustering offers: (1) an *optimal* compression approach for any chosen K, thus meeting the requirement of good representation of the original SCIRD-TS space; (2) the desirable high mutual incoherence (i.e. high inter-cluster Euclidean distance); (3) a fast compression algorithm (run time negligible compared to the CSC phase). So, if we indicate with $\mathbf{f}^{(i)}$ the i-th SCIRD-TS filter in $\overline{\mathcal{F}}$ ($\mathbf{f}^{(i)}$ is $F(\boldsymbol{x}; \boldsymbol{\sigma}, k, \theta)$ in Eq. (4) arranged as a one-column vector), the second step of the proposed warm-start strategy consists of solving the optimization problem

$$\underset{D,\mathbf{c}}{\operatorname{argmin}} \sum_i \left\| D\mathbf{c}^{(i)} - \mathbf{f}^{(i)} \right\|_2^2 \tag{6}$$

subject to $\|\mathbf{c}^{(i)}\|_0 \leq 1, \forall i = 1, \ldots, m_D$ and $\|\mathbf{d}^{(j)}\|_2 = 1, \forall j = 1, \ldots, K$, where $\mathbf{c}^{(i)}$ is the code vector related to the i-th original SCIRD-TS filter $\mathbf{f}^{(i)}$, and $\mathbf{d}^{(j)}$ is the j-th column of the dictionary D of prototype filters (examples in Figure 6-first column). In our experiments, we adopt the fast K-means optimisation algorithm proposed by Coates and Ng in [36][3]. Careful seeding discussed in [37] is used to initialise the clusters.

B. Refining the prototype filters by CSC

We refine the filter bank obtained with the warm-start strategy by CSC. Specifically, we optimise the following objective function [11]:

$$\underset{\substack{\{D^{(j)}\} \\ \{M_i^{(j)}\}}}{\operatorname{argmin}} \sum_{i=1}^{N} \left(\left\| P_i - \sum_{j=1}^{K} D^{(j)} * M_i^{(j)} \right\|_2^2 + \lambda \sum_{j=1}^{K} \left\| M_i^{(j)} \right\|_1 \right), \tag{7}$$

where P_i is the i-th original image patch to reconstruct (N patches in total), $D^{(j)}$ is the j-th refined filter (K filters in total), $M_i^{(j)}$ can be regarded as the j-th component (map) of the representation related to P_i and λ is the sparsity (regularization) parameter. Filters, original image patches and representation maps are arranged as matrices. The symbol $*$ indicates convolution.

In essence, the goal of this CSC optimisation is to minimise the total reconstruction error computed by approximating each original image patch using the current filter bank. The reconstruction is obtained by finding a sparse representation (the second term in Eq. (7) penalises the ℓ_1-norm of each component of the representation) of the current patch. Since the objective in Eq. (7) is not convex, several (sub-optimal) optmisation strategies can be employed. For instance, Rigamonti et al. [11] adopted a proximal algorithm, i.e. *ISTA* (iterative shrinkage thresholding algorithm) [38], [39]. To speed-up the optimisation, we adopt a faster proximal method, i.e. *FISTA* [35]. Moreover, we compute the high number of convolutions in the Fourier domain by exploiting fast Fourier transform algorithms. Finally, we adopt a batch-based optimisation strategy as done, for instance, in [40], [41].

C. Impact of the warm-start strategy on CSC optimisation

We provide a brief analysis of the computational complexity of CSC optimisation, in terms of number of multiplications, to better investigate the impact of the proposed warm-start strategy on the running time.

Let $I_1 \in \mathbb{R}^{r_1 \times c_1}$ and $I_2 \in \mathbb{R}^{r_2 \times c_2}$ be two images (or patches) we want to convolve. Due to the high number of convolutions involved in the CSC optimisation, we compute them in the Fourier domain, hence requiring the following steps:

1) Padding I_1 and I_2 with zeros so that they have the same size $r_3 \times c_3$, where r_3 and c_3 are the closest powers of 2 larger than $r_1 + r_2 - 1$ and $c_1 + c_2 - 1$, respectively;

[2]We adopt the same distance used for the CSC phase.

[3]Notice that this algorithm does not guarantee convergence to the global minimum but to a local one, so the compression is *locally* optimal.

2) Computing the Fourier transform (DFT) of the two images;
3) Multiplying the DFTs of the two images;
4) Computing the inverse Fourier transform (IDFT) of the result.

Considering that a DFT (and also an IDFT) requires $6r_3c_3\log_2(r_3c_3)$ real multiplications [16], and that a complex multiplication requires 3 real multiplications, a single convolution would require $3r_3c_3(6\log_2(r_3c_3)+1)$ multiplications.

The fast proximal method (FISTA) we adopt to optimise Eq. (7) alternates between the optimisation w.r.t. the K filters ($D^{(j)}$) and the maps ($M_i^{(j)}$) for each patch P_i (refer to the MATLAB implementation for details):

Optimisation w.r.t. the filters. This can be obtained by gradient descent, which amounts to computing K convolutions between the residual error of reconstruction and the related K maps, as the second term of Eq. (7) vanishes [42]. The total number of multiplications needed to perform this step is therefore[4] $3Kr_3c_3(6\log_2(r_3c_3)+1)$.

Optimisation w.r.t. the maps. From a computational complexity perspective (refer to the MATLAB implementation for details), this step requires the computation of the gradient of the first term in Eq. (7) w.r.t. the maps $M_i^{(j)}$ and a soft-thresholding (proximal operator of the l_1 norm [39], [42]). Again, the gradient can be computed efficiently by convolving the K filters with the residual error of reconstruction [42], hence requiring $3Kr_3c_3(6\log_2(r_3c_3)+1)$ multiplications. In addition, the K soft-thresholding operations require Kr_3c_3 multiplications [30].

Since we optimise over N patches (also called "mini-batch" in batch-based optimisation strategies [40], [41], [43]) and iterate several times (every pass over all the N patches is denoted as "epoch", N_e), the total number of multiplications required to optimise Eq. (7) is:

$$N_e \times N \times [6Kr_3c_3(6\log_2(r_3c_3)+1) + Kr_3c_3]. \quad (8)$$

The number of patches N, the number of filters K and the dimension of the filters are application-dependent. Once the optimisation algorithm is fixed (FISTA, in our case), the only other parameter which could have a significant impact on the complexity is the number of epochs N_e (multiplicative factor). We demonstrate experimentally in Section IV that initialising CSC with our warm-start strategy reduces N_e (and often achieves lower reconstruction errors, thus potentially leading to more discriminative filter banks).

III. Data Sets and Evaluation Criteria

We employed four benchmark data sets to validate the proposed CSC acceleration strategy. They include two of the most used public data sets to validate retinal blood vessel segmentation, DRIVE [3] and STARE [2], and two data sets showing neurites, BF2D and VC6, used as benchmark in recent work [11], [13], [15], [16], [32]. In this section, we first describe the data sets (visual examples in Figure 8) and discuss the evaluation criteria. Then, we discuss and report the adopted parameters setting.

[4]We could pre-compute the DFT of the residual error and reduce the number of multiplications further.

A. Data sets

DRIVE[5] [3] has been widely adopted as benchmark data set for vessel segmentation [4]–[6], [8]–[12], [16]–[20]. It includes 40 colour retinal images from a diabetic retinopathy screening program in the Netherlands. The images were acquired by a fundus camera (CR5 non-mydriatic 3-CCD, Canon, Tokio, Japan) with 45 degrees field of view. Each image is 768×584 pixels. The data set was originally split in training and testing set in [3], each including 20 images, and we adopted the same set partition. Manual segmentations were generated by two different specialists for each image. Following the literature (e.g. [4], [10]), we adopted the first observer as ground truth.

BF2D was first used by the authors in [11]. It consists of two minimum projections of bright-field micrographs that capture neurons. The images have high resolution; their size is 1024×1792 and 768×1792 pixels. We adopted the same set partition described by the authors. The data set includes masks to eliminate the nucleus and manual segmentations generated by an expert.

VC6 was created by the authors also in [11] from a set of 3D images showing dendritic and axonal subtrees from one neuron in the primary visual cortex. The original 3-D images are part of the publicly available data set used recently for the international DIADEM segmentation challenge (Visual Cortical Layer 6 Neuron) [24]. This data set includes three high-resolution images (882×378, 630×441 and 817×588 pixels) obtained by computing minimum intensity projections of three image stacks (3-D images). We adopted the same set partition used by the authors, using two images for training and retaining the third for testing. The data set includes manual segmentations provided by experts.

STARE[6] [2] is another data set including fundus images, widely used as benchmark for retinal vessel segmentation [4]–[6], [9]–[11], [18]–[20]. The full data set includes 397 colour images captured by a TopCon TRV-50 fundus camera at 35 degrees field of view. Each image is 605×700 pixels. A subset of 20 images (10 normal and 10 abnormal) were manually segmented by two experts [2]. Following the literature (e.g. [4], [10]), we adopted the first observer as ground truth.

Poor and variable contrast, low-resolution, non-uniform illumination, structure fragmentation, irregularities in the staining process (VC6), confounding non-target structures (e.g. optic disk, exudates and haemorrhages in DRIVE and STARE; blob-like structures in BF2D and VC6) make these data sets particularly challenging for automatic segmentation.

B. Performance Evaluation

Since the CSC optimisation problem aims to find a sparse representation for each original image patch minimising the total reconstruction error, we first assessed the performance in terms of reconstruction error and time to convergence. Then, we evaluated segmentation performance.

Reconstruction error and time to convergence. For these experiments, we randomly sampled $1,000$ 49×49 pixels

[5]http://www.isi.uu.nl/Research/Databases/DRIVE/
[6]http://www.ces.clemson.edu/~ahoover/stare/

original image patches (i.e. the "batch") from the training set of DRIVE, BF2D and VC6 separately and measured the total reconstruction error against the number of epochs[7]. For STARE, we excluded the 20 manually segmented images used for assessing segmentation performance and carried out this experiment on the 377 images left. We compare the performance of our initialisation strategy against the random (adopted in most of the related work, e.g. [11], [16], [30], [31]) and DCT-based (adopted in [33], [34]) one. We used the same batch for the proposed method and the baselines, for fair comparison. To assess the influence of the dictionary size on the total reconstruction error we ran experiments for banks including 49, 100 and 144 learned filters.

Segmentation. To assess segmentation performance, we convolve each image with the K learned filters and represent each pixel with the K local responses (i.e. K-D feature vector). Then, we give this feature vector as input to a random forest classifier to infer the probability of each pixel of belonging to a curviliear structure. For DRIVE, BF2D and VC6, the training set was formed by pixel samples from the provided training images; for STARE, we adopted a leave-one-out cross-validation on the 20 images manually segmented, as typically done in the literature (e.g. [4], [20]). We adhere to the evaluation protocol adopted in [11], [13], [15], [16], [23], [32], among others. Specifically, given the noticeable imbalance between true negatives (TNs) and the other measures of the contingency matrix, i.e. true positives (TPs), false negatives (FNs) and false positives (FPs)[8], we adopt precision-recall (PR) curves and area under PR curves (AUPRC) to assess segmentation performance. In addition to the baselines adopted above (i.e. CSC with random and DCT initialisation), we compare the proposed method performance with widely used HCFs (i.e. Gabor [4], Frangi [1], OOF [7]), SCIRD [13], [23] and the combination method proposed by Rigamonti et al. [11].

C. Parameters Setting

HCFs. Parameters for SCIRD-TS and baseline methods were tuned separately to achieve their best performance on each data set, to provide a fair comparison.

Warm-start strategy. Parameters ranges for generating the large SCIRD-TS filter bank were set manually by visually inspecting DRIVE training images, with the idea of covering a suitable range in terms width, elongation, curvature and rotation resolution. We adopted a conservative setting (i.e., wide ranges and high resolution) without careful tuning or specific optimisation. In particular, $\sigma_1 = [1, 10]$ with step 0.5, $\sigma_2 = [1, 10]$ with step 0.5 (filters are forced to be elongated, i.e. filters with $\sigma_2 > \sigma_1$ are discarded), $k = [-0.1, 0.1]$ with step 0.025 and $\theta = [15, 180]$ with step 15 degrees. To test the generalisation of this setting, we adopted it for BF2D, VC6 and STARE as well, although they show different curvilinear structures (neurites vs retinal blood vessels) resolution and non-target structures. We set the number of K-means iterations to 100, although a few tens are typically sufficient (negligible impact on the total time to convergence of the proposed acceleration strategy). We assessed the influence of the number of filters (i.e., K) on the reconstruction performance, and used $K = \{49, 100, 144\}$. For comparison, the maximum number of CSC filters learned in [17] (the current benchmark on DRIVE) is 121.

[7]In batch-based optimisation strategies, an epoch represents one pass over the entire batch.

[8]The number of true background pixels is much higher than that of true vessel or neurite pixels in the images.

TABLE I
TOTAL TIME TO CONVERGENCE (IN MINUTES) FOR THE CSC PHASE INITIALISED WITH OUR METHOD, AND THE BASELINES. IN BRACKETS, OUR WARM START PROCESSING TIME (IN SECONDS).

DRIVE	**Number of learned filters**		
Method	**49**	**100**	**144**
Random	167'	458'	1085'
DCT	167'	242'	2049'
Ours	51'(6")	106'(11")	195'(16")

BF2D	**Number of learned filters**		
Method	**49**	**100**	**144**
Random	152'	209'	1062'
DCT	198'	418'	1474'
Ours	58'(6")	141'(11")	247'(16")

VC6	**Number of learned filters**		
Method	**49**	**100**	**144**
Random	132'	374'	467'
DCT	345'	734'	743'
Ours	54'(6")	117'(11")	203'(16")

STARE	**Number of learned filters**		
Method	**49**	**100**	**144**
Random	120'	291'	466'
DCT	313'	397'	751'
Ours	61'(6")	94'(11s)	159'(16")

CSC phase. When random initialisation is used, setting the sparsity parameter λ manually is not trivial. In fact, low values tend to produce noisy filters, whereas high ones lead to a slow convergence. We found $\lambda = 2$ to yield good results on the DRIVE data set (we investigated the impact of different λ values and report the results below). To test robustness, we used the same setting (i.e. $\lambda = 2$) for BF2D, VC6 and STARE as well.

Classifier. We trained a random decision forest [44], [45] (henceforth, RF) using 144-D feature vectors (i.e. number of learned filters $K = 144$) with 100 trees for each data set, to achieve a good compromise between segmentation performance and processing time. Each tree's depth was set automatically, by evaluating the out-of-bag error during training. We randomly sampled $200,000$ training instances from the training partition of each data set to build the related RF model. We adopted the same filter size used in [11], [16], [17], i.e. 21×21 pixels for all the data sets, as size was not found to affect performance significantly on the same data sets.

All the experiments were carried out on a laptop equipped with Intel i7-4702 CPU at 2.2GHz and 16GB RAM (MATLAB implementations).

IV. EXPERIMENTS AND RESULTS

A. Reconstruction error and time to convergence

In Table I we report the total time to convergence for CSC using our acceleration method and the baselines, for each data

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMI.2016.2570123, IEEE Transactions on Medical Imaging

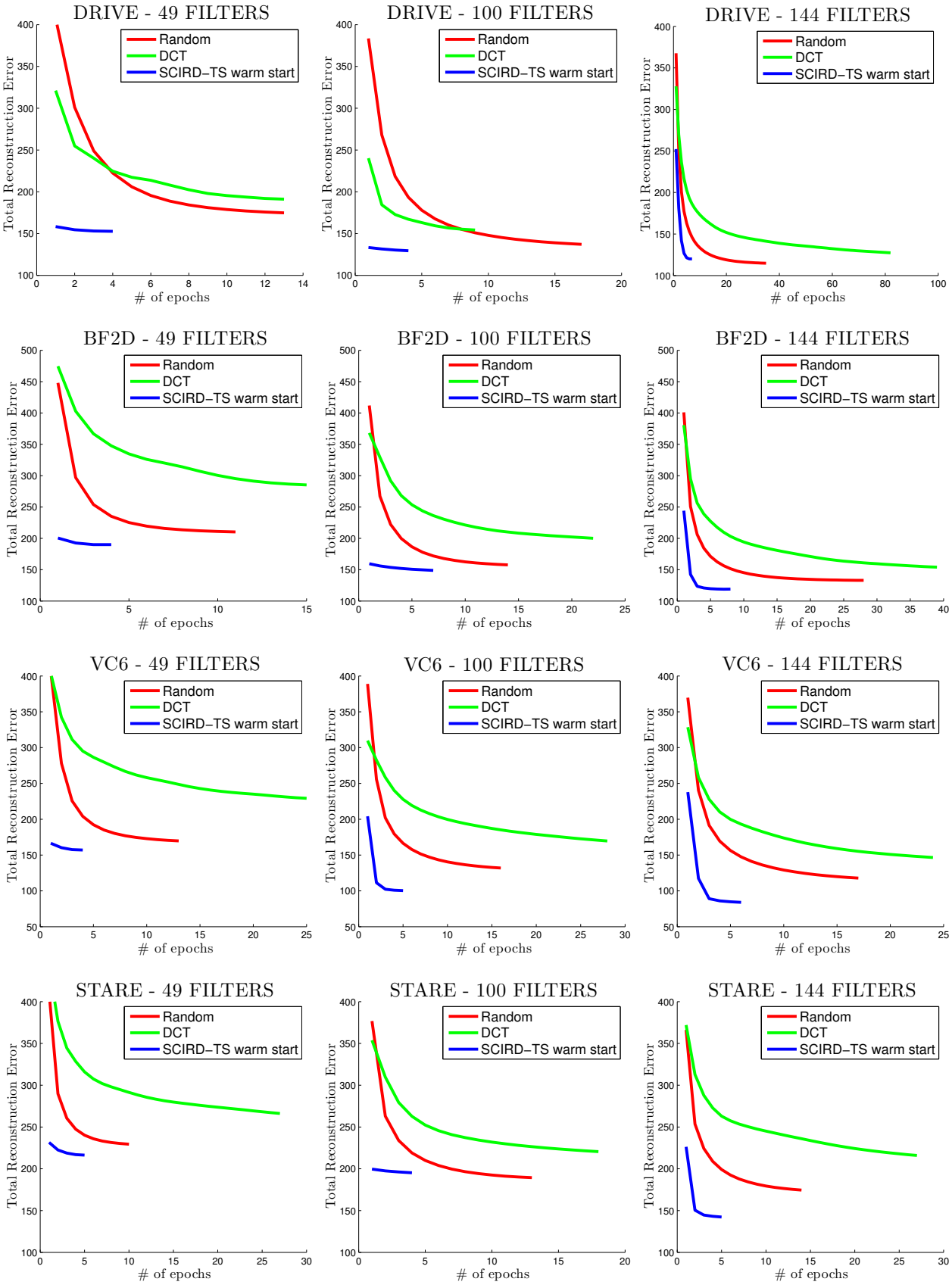

Fig. 5. **Experiments: reconstruction error and time to convergence.** Performance evaluation in terms of total reconstruction error for CSC with random, DCT and SCIRD-TS initialisation. Each row shows the influence of the dictionary size on the total reconstruction error, for each data set. Optimisations were stopped at convergence. Notice that the proposed initialisation approach ("SCIRD-TS warm start") achieves the lowest reconstruction error for each filter size, and simultaneously has much faster convergence, compared to conventional initialisations.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMI.2016.2570123, IEEE Transactions on Medical Imaging

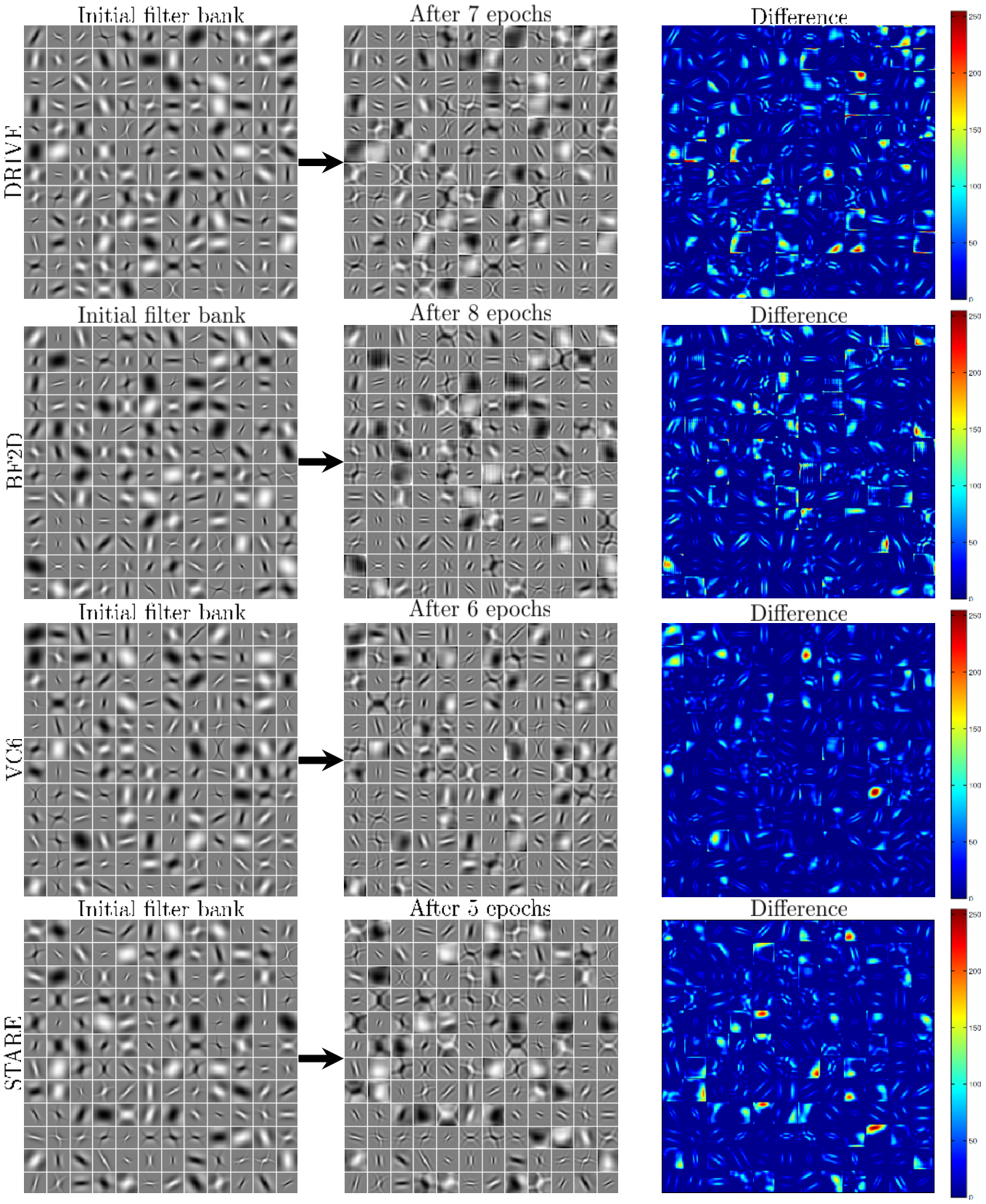

Fig. 6. **Visualisation of a CSC-refined SCIRD-TS filter bank.** SCIRD-TS filter banks obtained after the fast warm-start strategy (first column), refinement by CSC (second column) and difference (third column) for DRIVE, BF2D and VC6 (refer to Table I for time to convergence). Some of the original filters are unchanged, while most of the others are only modified in length or width.

set and dictionary size. We observe that (1) the time to run our warm-start strategy is negligible compared to the total time to run CSC (i.e. a few seconds against tens of minutes); (2) the proposed CSC acceleration takes much less time to obtain discriminative filter banks than conventional CSC initialisation strategies, e.g. up to 82% less time, when 144 filters are learned. Remarkably, our acceleration strategy does not compromise performance either in terms of reconstruction error or segmentation performance. Figure 5 shows the total reconstruction error against the number of epochs needed to achieve convergence for our initialisation method and the baselines, for the four data sets and different dictionary size. We notice that (1) our warm-start strategy based on SCIRD-TS achieves both the lowest total reconstruction error and the fastest convergence on each data set and for each dictionary size[9]; (2) initialising the filter bank with DCT (as done in [33], [34]) does not lead to either faster convergence or lower reconstruction error compared to random initialisation, for data sets including curvilinear structures; (3) although the adopted SCIRD-TS parameters were set using DRIVE training images, the total reconstruction error on BF2D, VC6 and STARE is always lower, and sometimes substantially, than random initialisation, thus suggesting good generalisation.

Figure 6 shows how the initial filter banks generated using the proposed warm-start strategy were refined by the adopted CSC approach on each data set (for compactness, only the largest filter banks are shown). A large subset of filters is left unchanged or refined lightly (in terms of width and elongation, for instance), while other filters are modified significantly to reduce the reconstruction error and compensate for the part HCFs are not capable to model. This observation confirms our hypothesis that a well-designed HCF bank includes already a large portion of the filters suitable for curvilinear structures segmentation in the medical domain, and that our approach (optimal warm-start) obtains highly discriminative filter banks in a more efficient way (compared to conventional initialisation).

B. Segmentation

Figure 7 shows the segmentation performance on the four data sets in terms of precision-recall curves for state-of-the-art and widely used HCFs (i.e. Gabor [4], Frangi [1], OOF [7]), SCIRD [13], [23], the proposed SCIRD-TS, the combination approach proposed by Rigamonti et al. [11] and CSC initialised with random (as done by [11], [16], [30]–[32]), DCT (as done by [33], [34]) and the proposed warm-start strategy. First, the proposed SCIRD-TS outperforms SCIRD [13], [23] (and the other HCFs baselines) on the four data sets, as it detects thinner structures not detected by SCIRD. Second, due to their modelling limitations and suboptimal parameter setting, HCFs are outperformed by methods based on discriminative filter learning.

Remarkably, precision-recall curves suggest that our acceleration strategy leads to filter banks matching or even exceeding the segmentation performance of CSC strategies initialised randomly or with general purpose HCFs (i.e. DCT), while converging in much less time. This is confirmed by quantitative results in terms of AUPRC, F-measure, Jaccard Index (aka Intersection Over Union, or IOU) and time needed to converge reported in Table II. Qualitative comparisons (probability maps) with the best performing baseline (i.e. random initialisation) are reported in Figure 8.

[9]In Figure 5, random initialisation achieves slightly less reconstruction error on "DRIVE - 144 filters", with a substantially higher number of epochs.

TABLE II
COMPARISON IN TERMS OF AUPRC, F-MEASURE, JACCARD INDEX AND TRAINING TIME (IN MINUTES), BETWEEN RANDOM, DCT-BASED AND THE PROPOSED INITIALISATION STRATEGY (DENOTED AS "OURS").

DRIVE	**Performance measure**			
Method	**AUPRC**	**F-measure**	**Jaccard**	**Time**
Random	0.85	0.77	0.62	1085'
DCT	0.84	0.76	0.61	2049'
Ours	**0.87**	**0.79**	**0.64**	**195'**
BF2D	**Performance measure**			
Method	**AUPRC**	**F-measure**	**Jaccard**	**Time**
Random	0.83	**0.77**	**0.62**	1062'
DCT	0.83	0.76	0.61	1474'
Ours	**0.84**	0.76	**0.62**	**247'**
VC6	**Performance measure**			
Method	**AUPRC**	**F-measure**	**Jaccard**	**Time**
Random	0.81	0.74	0.59	467'
DCT	0.77	0.70	0.54	743'
Ours	**0.83**	**0.76**	**0.62**	**203'**
STARE	**Performance measure**			
Method	**AUPRC**	**F-measure**	**Jaccard**	**Time**
Random	0.84	0.75	0.58	466'
DCT	0.83	0.74	0.57	751'
Ours	**0.86**	**0.77**	**0.60**	**159'**

TABLE III
INFLUENCE OF THE SPARSITY PARAMETER λ ON THE SEGMENTATION PERFORMANCE (AUPRC) OF THE PROPOSED INITIALISATION METHOD (*Ours*) AND THE BEST BASELINE METHOD (*Random*) ON DRIVE.

	λ		
Init. method	**0.2**	**2**	**20**
Random	0.8418	0.8515	0.8461
Ours	**0.8638**	**0.8676**	**0.8655**

We investigated the influence of the sparsity parameter (λ) on the segmentation performance when the random and the proposed initialisation strategy are employed. Specifically, we repeated the experiments with λ decreased and increased by a factor 10 compared to the adopted setting (i.e. $\lambda = 0.2$ and $\lambda = 20$, respectively) on DRIVE data set. Experimental results (Table III) suggest that CSC initialised with our warm-start strategy is more robust against this critical parameter setting, compared to random initialisation, an important advantage in terms of adaptation to different data sets (if confirmed by future experiments).

It is worth noting that our segmentation pipeline is single-layer, yet it achieves the same level of performance of the multi-layer architecture proposed by Sironi et al. [32] on DRIVE (F-measure = 0.79); the latter is based on CSC filter banks leveraged by an auto-context regression pipeline recently improved by a post-processing strategy and shown to achieve state-of-the-art segmentation performance [17]. However, the authors report that learning a different convolutional

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMI.2016.2570123, IEEE Transactions on Medical Imaging

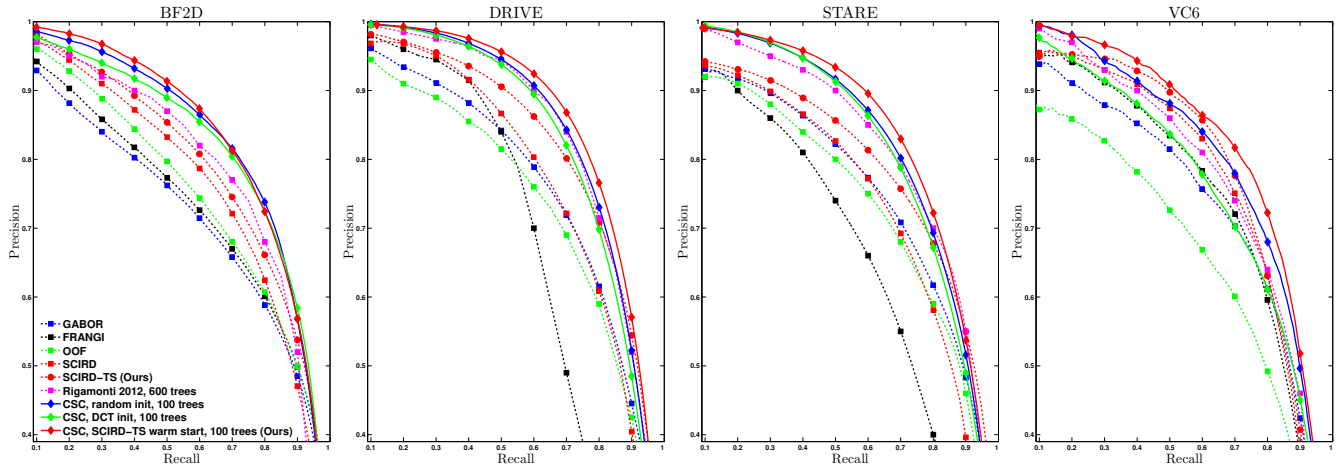

Fig. 7. **Experiments: segmentation.** Performance evaluation in terms of precision-recall curves for pixel-level segmentation. Notice that we employed a RF with only 100 trees, compared to the method proposed by Rigamonti et al. [11] in which 600 trees were used, hence slower at testing time.

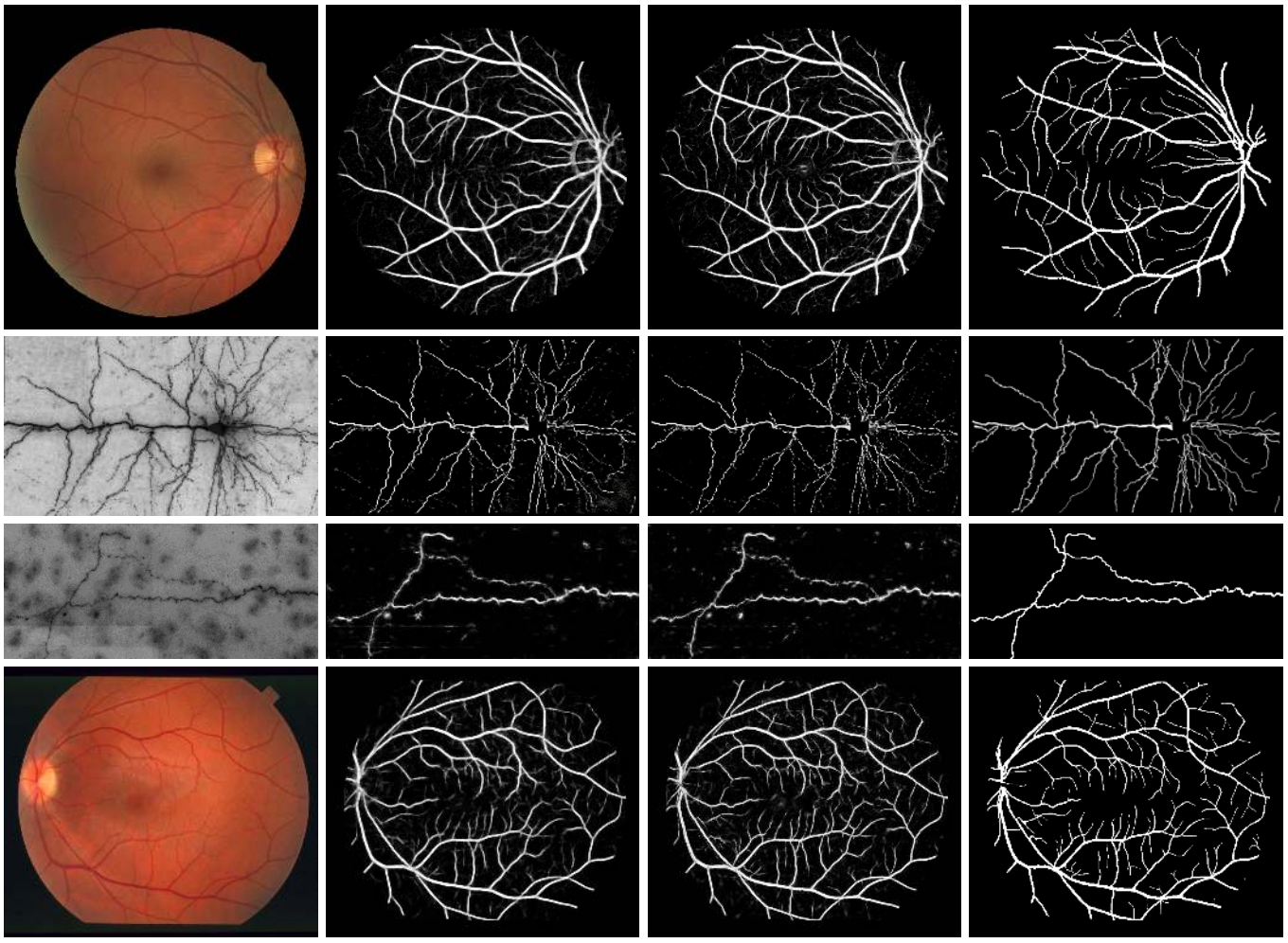

Fig. 8. **Experiments: segmentation.** Probability maps computed on images from DRIVE (first row), BF2D (second row), VC6 (third row) and STARE (fourth row). For each row, from left to right, we report original image, result of the best performing baseline (i.e. "CSC, random init."), proposed method's result and ground truth.

filter bank for each layer of this auto-context architecture is *prohibitively expensive* [32], hence they learn a single filter bank (121 filters) and use it for all the layers. Given the speed-up obtained by using the proposed acceleration strategy (without performance degradation for reconstruction and segmentation), (1) a convolutional filter bank could be

learned for each layer to model higher-order properties of curvilinear structures and potentially improve segmentation performance; (2) alternatively, the proposed acceleration strategy could significantly reduce its training time and therefore speed-up adaptation to other data sets.

V. Discussion and Conclusions

CSC is a central machine learning strategy in current state-of-the-art approaches to curvilinear structure segmentation in the medical domain (e.g. [17]). Its main drawback is that learning large filter banks is very time consuming [32], and accelerating CSC has recently become a particularly active area of research. Previous approaches have focussed on the optimisation itself (e.g. [31]) and important advances have been made. We address acceleration from a different perspective, i.e. initialisation, with the important benefit of being complementary to approaches focussing on the optimisation. Driven by the observation that filter banks obtained by CSC applied to curvilinear structures often incorporate filters closely resembling hand-crafted ones, we have proposed and tested a novel approach to accelerate CSC based on carefully designed HCFs and an optimal (fast) warm-start strategy. Our approach obtains incoherent dictionaries from a large set of filters generated by sampling uniformly and densely the adopted HCFs ranges.

We have tested the performance of the proposed method using two quantitative strategies: total reconstruction error and segmentation performance.

Experiments measuring the reconstruction error of random batches from four diverse data sets show that (1) CSC accelerated with our warm-start strategy generates filter banks much faster (e.g., up to 82% less time, when 144 filters are learned) compared to conventional initialisation strategies (i.e. random or DCT); (2) remarkably, the speed-up does not degrade performance: the reconstruction error is often substantially lower than that of the baselines; (3) adopting HCFs designed for curvilinear structure segmentation (e.g. SCIRD-TS) is crucial, as the general purpose DCT does not lead to neither a speed-up, nor lower reconstruction error on data sets including curvilinear structures, as aspect never teased out before, to our knowledge.

Experiments on the target application, i.e. curvilinear structure segmentation, suggest that (1) HCFs alone are considerably outperformed by methods based on discriminative filter learning, due to modelling limitations and sub-optimal (manual) parameter setting; (2) our strategy to accelerate CSC outperforms random and DCT initialisation on DRIVE, STARE and particularly on VC6, whereas it matches the performance of random initialisation on BF2D (requiring 77% less training time).

We have tested the parameter setting generalisation by adopting the same setting used for DRIVE also for BF2D, VC6 and STARE data sets, although the latter include different curvilinear structures (retinal blood vessels in DRIVE and neurites in BF2D, VC6) and different image characteristics in terms of contrast and confounding structures (STARE includes 10 abnormal images).

Experiments show that most of the prototype filters obtained by the warm-start strategy change lightly (i.e. only in terms of width, elongation, curvature), while others more noticeably. Although we expect that the latter may have a more positive impact on the segmentation performance than the former, future experiments will be carried out to investigate this aspect further.

We reckon that combining our acceleration strategy, based on initialisation, with the ones based on optimisation can lead to a substantial speed-up of CSC-based curvilinear structure segmentation approaches, yielding even faster training and, therefore, faster adaptation to different data sets and/or better segmentation performance, by enabling the learning of larger filter banks. This is the research direction we will explore in the near future.

Acknowledgments

The authors are grateful to the CVlab (EPFL, CH) for providing VC6 and BF2D data sets. They would like to thank S. McKenna and J. Zhang (CVIP, University of Dundee, UK) for valuable comments. They are also grateful to the anonymous reviewers for their valuable comments and suggestions, which contributed to improve this paper.

References

[1] A. Frangi, W. Niessen, K. Vincken, and M. Viergever, "Multiscale vessel enhancement filtering," *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 130–137, 1998.

[2] A. Hoover, V. Kouznetsova, and M. Goldbaum, "Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response," *IEEE Transactions on Medical Imaging*, vol. 19, no. 3, pp. 203–210, 2000.

[3] J. Staal, M. Abramoff, M. Niemeijer, M. Viergever, and B. van Ginneken, "Ridge-based vessel segmentation in color images of the retina," *IEEE Transactions on Medical Imaging*, vol. 23, pp. 501–509, Apr. 2004.

[4] J. Soares, J. Leandro, R. Cesar, H. Jelinek, and M. Cree, "Retinal vessel segmentation using the 2-d gabor wavelet and supervised classification," *IEEE Transactions on Medical Imaging*, vol. 25, pp. 1214–1222, Sep. 2006.

[5] A. Mendonca and A. Campilho, "Segmentation of retinal blood vessels by combining the detection of centerlines and morphological reconstruction," *IEEE Transactions on Medical Imaging*, vol. 25, no. 9, pp. 1200–1213, Sep. 2006.

[6] E. Ricci and R. Perfetti, "Retinal blood vessel segmentation using line operators and support vector classification," *Medical Imaging, IEEE Transactions on*, vol. 26, no. 10, pp. 1357–1365, 2007.

[7] M. W. Law and A. C. Chung, "Three dimensional curvilinear structure detection using optimally oriented flux," in *Computer Vision–ECCV 2008*. Springer, 2008, pp. 368–382.

[8] B. Al-Diri, A. Hunter, and D. Steel, "An active contour model for segmenting and measuring retinal vessels," *IEEE Transactions on Medical Imaging*, vol. 28, pp. 1488–1497, Sep. 2009.

[9] B. Lam, Y. Gao, and A.-C. Liew, "General retinal vessel segmentation using regularization-based multiconcavity modeling," *IEEE Transactions on Medical Imaging*, vol. 29, no. 7, pp. 1369 –1381, Jul. 2010.

[10] D. Marin, A. Aquino, M. E. Gegundez-Arias, and J. M. Bravo, "A new supervised method for blood vessel segmentation in retinal images by using gray-level and moment invariants-based features," *IEEE Transactions on Medical Imaging*, vol. 30, pp. 146–158, Jan. 2011.

[11] R. Rigamonti and V. Lepetit, "Accurate and efficient linear structure segmentation by leveraging ad hoc features with learned filters." in *MICCAI (1)*, ser. Lecture Notes in Computer Science, vol. 7510. Springer, 2012, pp. 189–197.

[12] Y. Ganin and V. Lempitsky, "N4-fields: Neural network nearest neighbor fields for image transforms," in *Computer Vision–ACCV 2014*. Springer, 2014, pp. 536–551.

[13] R. Annunziata, A. Kheirkhah, P. Hamrah, and E. Trucco, "Scale and curvature invariant ridge detector for tortuous and fragmented structures," in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2015*, ser. Lecture Notes in Computer Science, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Springer International Publishing, 2015, vol. 9351, pp. 588–595.

[14] R. Annunziata, A. Garzelli, L. Ballerini, A. Mecocci, and E. Trucco, "Leveraging multiscale hessian-based enhancement with a novel exudate inpainting technique for retinal vessel segmentation," *IEEE Journal of Biomedical and Health Informatics*, vol. PP, no. 99, 2015.

[15] R. Annunziata, A. Kheirkhah, P. Hamrah, and E. Trucco, "Boosting hand-crafted features for curvilinear structure segmentation by learning context filters," in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2015*, ser. Lecture Notes in Computer Science, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Springer International Publishing, 2015, vol. 9351, pp. 596–603.

[16] A. Sironi, B. Tekin, R. Rigamonti, V. Lepetit, and P. Fua, "Learning separable filters," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 37, no. 1, pp. 94–106, 2015.

[17] A. Sironi, V. Lepetit, and P. Fua, "Projection onto the manifold of elongated structures for accurate extraction," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 316–324.

[18] L. Gu and L. Cheng, "Learning to boost filamentary structure segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 639–647.

[19] Y. Zhao, L. Rada, K. Chen, S. Harding, and Y. Zheng, "Automated vessel segmentation using infinite perimeter active contour model with hybrid region information with application to retinal images," *Medical Imaging, IEEE Transactions on*, vol. 34, no. 9, pp. 1797–1807, Sept 2015.

[20] Q. Li, B. Feng, L. Xie, P. Liang, H. Zhang, and T. Wang, "A cross-modality learning approach for vessel segmentation in retinal images," *Medical Imaging, IEEE Transactions on*, vol. 35, no. 1, pp. 109–118, Jan 2016.

[21] E. Trucco, A. Ruggeri, T. Karnowski, L. Giancardo, E. Chaum, J. P. Hubschman, B. al Diri, C. Y. Cheung, D. Wong, M. Abrmoff, G. Lim, D. Kumar, P. Burlina, N. M. Bressler, H. F. Jelinek, F. Meriaudeau, G. Quellec, T. MacGillivray, and B. Dhillon, "Validating retinal fundus image analysis algorithms: Issues and a proposalvalidating retinal fundus image analysis algorithms," *Investigative Ophthalmology & Visual Science*, vol. 54, no. 5, p. 3546, 2013.

[22] R. Annunziata, A. Kheirkhah, S. Aggarwal, B. M. Cavalcanti, P. Hamrah, and E. Trucco, "Two-dimensional plane for multi-scale quantification of corneal subbasal nerve tortuositymulti-scale quantification of corneal nerve tortuosity," *Investigative Ophthalmology & Visual Science*, vol. 57, no. 3, p. 1132, 2016.

[23] R. Annunziata, A. Kheirkhah, S. Aggarwal, P. Hamrah, and E. Trucco, "A fully automated tortuosity quantification system with application to corneal nerve fibres in confocal microscopy images," *Medical Image Analysis*, vol. 32, pp. 216 – 232, 2016.

[24] K. M. Brown, G. Barrionuevo, A. J. Canty, V. De Paola, J. A. Hirsch, G. S. Jefferis, J. Lu, M. Snippe, I. Sugihara, and G. A. Ascoli, "The diadem data sets: representative light microscopy images of neuronal morphology to advance automation of digital reconstructions," *Neuroinformatics*, vol. 9, no. 2-3, pp. 143–157, 2011.

[25] K. Kavukcuoglu, P. Sermanet, Y.-L. Boureau, K. Gregor, M. Mathieu, and Y. L. Cun, "Learning convolutional feature hierarchies for visual recognition," in *Advances in neural information processing systems*, 2010, pp. 1090–1098.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[27] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 8, pp. 1798–1828, 2013.

[28] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[29] P. Kontschieder, M. Fiterau, A. Criminisi, and S. R. Bulo', "Deep neural decision forests," in *Intl. Conf. on Computer Vision (ICCV), Santiago, Chile*, December 2015.

[30] H. Bristow, A. Eriksson, and S. Lucey, "Fast convolutional sparse coding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 391–398.

[31] F. Heide, W. Heidrich, and G. Wetzstein, "Fast and flexible convolutional sparse coding," in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*. IEEE, 2015, pp. 5135–5143.

[32] A. Sironi, E. Turetken, V. Lepetit, and P. Fua, "Multiscale centerline detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[33] C. Bao, H. JI, Y. Quan, and Z. Shen, "Dictionary learning for sparse coding: Algorithms and analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[34] C. Bao, Y. Quan, and H. Ji, *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI*. Springer International Publishing, 2014, ch. A Convergent Incoherent Dictionary Learning Algorithm for Sparse Coding, pp. 302–316.

[35] 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.

[36] A. Coates and A. Y. Ng, "Learning feature representations with k-means," in *Neural Networks: Tricks of the Trade*. Springer, 2012, pp. 561–580.

[37] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[38] F. Bach, R. Jenatton, J. Mairal, G. Obozinski *et al.*, "Convex optimization with sparsity-inducing norms," *Optimization for Machine Learning*, vol. 5, 2011.

[39] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, 2014.

[40] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Advances in neural information processing systems*, 2006, pp. 801–808.

[41] A. D. Szlam, K. Gregor, and Y. L. Cun, "Structured sparse coding via lateral inhibition," in *Advances in Neural Information Processing Systems*, 2011, pp. 1116–1124.

[42] R. Chalasani, J. C. Principe, and N. Ramakrishnan, "A fast proximal method for convolutional sparse coding," in *Neural Networks (IJCNN), The 2013 International Joint Conference on*. IEEE, 2013, pp. 1–5.

[43] J. Ngiam, A. Coates, A. Lahiri, B. Prochnow, Q. V. Le, and A. Y. Ng, "On optimization methods for deep learning," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 265–272.

[44] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[45] A. Criminisi, J. Shotton, and E. Konukoglu, "Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning," *Foundations and Trends® in Computer Graphics and Vision*, vol. 7, no. 2–3, pp. 81–227, 2012.