SEPARABLE DICTIONARY LEARNING FOR CONVOLUTIONAL SPARSE CODING VIA SPLIT UPDATES

Jorge Quesada†*, Paul Rodriguez*†*, and Brendt Wohlberg**

†Department of Electrical Engineering, Pontificia Universidad Catolica del Peru, Lima, Peru
*T-5 Applied Mathematics and Plasma Physics, Los Alamos National Laboratory, Los Alamos, NM, USA

ABSTRACT

The increasing ubiquity of Convolutional Sparse Representation techniques for several image processing tasks has recently sparked interest in the use of separable 2D dictionary filter banks (as alternatives to standard non-separable dictionaries) for efficient Convolutional Sparse Coding (CSC) implementations. However, existing methods approximate a set of K non-separable filters via a linear combination of R ($R << K$) separable filters, which puts an upper bound on the latter's quality. Furthermore, this implies the need to learn first the whole set of non-separable filters, and only then compute the separable set, which is not optimal from a computational perspective.

In this paper, we propose a method to directly learn a set of K separable dictionary filters from a given image training set by drawing ideas from standard Convolutional Dictionary Learning (CDL) methods. We show that the separable filters obtained by our method match the performance of an equivalent number of non-separable filters. Furthermore, the computational performance of our learning method is shown to be substantially faster than a state-of-the-art non-separable CDL method when either the image training set or the filter set are large.

***Index Terms*—** Convolutional Sparse Representation, Dictionary Learning, Separable filters

1. INTRODUCTION

Sparse representations and dictionary learning are well-known techniques in the field of signal and image processing, yielding effective approaches in tasks such as denoising, object recognition, and machine learning applications [1]. In particular, convolutional formulations, which model an image as a sum over a set of convolutions between coefficient maps and dictionary filters, have received increasing attention for their suitability to handle whole images, as opposed to their patch-based counterparts. In this formulation, the standard form of the Convolutional Sparse Coding (CSC) problem is given by the Convolutional Basis Pursuit Denoising (CBPDN) objective, namely:

$$\min_{\{\mathbf{x}_{k,s}\}} \frac{1}{2} \left\| \sum_{k=1}^{K} D_k * \mathbf{x}_k - \mathbf{b} \right\|_2^2 + \lambda \sum_{k=1}^{K} \|\mathbf{x}_k\|_1 \,, \tag{1}$$

where $\mathbf{b}$ is the observed image, $\{\mathbf{x}_k\}$ is the coefficient map set, and $\{D_k\}$ are the (non-separable) dictionary filters. The Convolutional Dictionary Learning extension is then given by:

$$\min_{\{D_k,\mathbf{x}_{k,s}\}} \frac{1}{2} \sum_{s=1}^{S} \left\| \sum_{k=1}^{K} D_k * \mathbf{x}_{k,s} - \mathbf{b}_s \right\|_2^2 + \lambda \sum_{s=1}^{S} \sum_{k=1}^{K} \|\mathbf{x}_{k,s}\|_1 \quad s.t. \quad \|D_k\|_2 = 1, \forall k, \tag{2}$$

where $\{\mathbf{b}_s\}$ is the set of training images (the constraint on the filter norms is needed to avoid scaling ambiguities).

It has been shown that using separable filters as dictionaries in tasks such as CSC [2] or Convolutional Neural Network (CNN) applications [3] provides significant improvements in computational performance with respect to non-separable implementations, with little loss in accuracy or reconstruction quality. In general, most of these methods rely on learning the separable filter set as an approximation of a previously obtained (usually large) set of non-separable filters, by using the equivalence:

$$D_k \approx \sum_{r=1}^{R} \alpha_{kr} G_r \quad k \in \{1, 2, \ldots, K\}, \tag{3}$$

which represents each non-separable filter $\{D_k\}$ as a linear combination of a smaller number of separable filters $\{G_r\}$ ($R << K$) [4]. This approach, however, depends heavily on the quality of the originating non-separable filters to obtain a good separable approximation. Furthermore, it implies a two step procedure: learning first the whole set of standard filters, and only then approximating the separable ones.

In this paper, we present an algorithm to directly learn the separable filters from the image training set by solving

$$\min_{\{h_r,v_r,\mathbf{x}_{r,s}\}} \frac{1}{2} \sum_{s=1}^{S} \left\| \sum_{r=1}^{R} v_r * h_r * \mathbf{x}_{r,s} - \mathbf{b}_s \right\|_2^2 + \lambda \sum_{s=1}^{S} \sum_{r=1}^{R} \|\mathbf{x}_{r,s}\|_1 \quad s.t. \quad \|h_r\|_2 = \|v_r\|_2 = 1, \forall r, \tag{4}$$

where $R = K$, and $\{h_r\}$ and $\{v_r\}$ are the horizontal and vertical components of each filter. The proposed method is derived as a natural extension of a well known CDL algorithm [5], and compared against both standard non-separable dictionaries and separable approximations learned via (3). Our computational results (see Section 4) show that the separable filters obtained by our method, when evaluated via a standard CBPDN problem, provide superior performance with respect to the approximated ones, while matching the same number of non-separable ones. Furthermore, the computational runtime of our learning algorithm is shown to be faster than standard non-separable learning approaches for most configurations.

Section 2.1 provides a brief review of the existing methods for dictionary learning in the literature, and Section 2.2 further details previous works on separable filter approximations. Section 3 presents the details of our proposed algorithm and relevant information regarding its efficient implementation.

2. PREVIOUS RELATED WORK

2.1. Non-separable (standard) dictionary learning

Since the CDL problem, as posed by (2), is non-convex when dealing with both variables ($\{\mathbf{x}_{k,s}\}$ and $\{D_k\}$) simultaneously, but becomes convex when keeping either of them constant, the most widely used minimization approach consists in alternating between the updates for the feature maps $\{\mathbf{x}_{k,s}\}$ (sparse coding) and the filters $\{D_k\}$ (dictionary learning). This section will address the main existing dictionary learning update methods (for a thorough review and comparison of sparse coding and dictionary learning updates and their coupling mechanisms, see [6]), which require solving a constrained convolutional form of the Method of Optimal Directions (MOD [7]), namely:

$$\min_{\{D_k\}} \frac{1}{2}\sum_{s=1}^{S}\left\|\sum_{k=1}^{K} D_k * \mathbf{x}_{k,s} - \mathbf{b}_s\right\|_2^2, \quad s.t. \quad \|D_k\|_2 = 1, \forall k, \tag{5}$$

for a given coefficient set $\{\mathbf{x}_{k,s}\}$.

Early methods solved this problem in the spatial domain, via variants of gradient descent [8] and MOD [9], among others [10, 11]. More recent implementations solve the most computationally demanding components of the problem in the frequency domain due to the associated speedup [6].

When performing the convolutions in the frequency domain, the filters must be zero-padded in order to have an adequate spatial support. This requirement can be denoted by a zero-padding projection operator P, and coupled with the normalization constraint into the constraint set:

$$C_{PN} = \{x \in \mathbb{R}^N : (I - PP^T)x = 0, \|x\|_2 = 1\}, \tag{6}$$

which allows to write the dictionary update in unconstrained form:

$$\min_{\{D_k\}} \frac{1}{2}\sum_{s=1}^{S}\left\|\sum_{k=1}^{K} D_k * \mathbf{x}_{k,s} - \mathbf{b}_s\right\|_2^2 + \sum_{r=1}^{R} \iota_{C_{\text{PN}}}(D_k), \tag{7}$$

where $\iota_{C_{\text{PN}}}(\cdot)$ is the indicator function of the constraint set C_{PN}. Several algorithms have been proposed to solve (7), most of which are based on Augmented Lagrangian frameworks, differing primarily on the approach they take to solve the ℓ_2 fidelity term subproblem. [12] proposed an Alternating Direction Method of Multipliers (ADMM [13]) formulation, which [5] and [14] later improved by efficiently approaching the aforementioned sub-problem using iterative Sherman Morrison and Matrix Inversion lemma solutions, respectively. Furthermore, [15] proposes an ADMM-consensus and a 3D formulation that decouple the problem from the number of training images S, thus improving the computational performance for the learning update.

There are also variants of these methods that perform the dictionary update in an online fashion such as [16] and [17], in order to save either computing time or memory resources during the learning process.

2.2. Separable from non-separable approximation

A straightforward approach to estimate G_r (as defined in Equation (3)) from a given set of standard filters $\{D_k\}$ was proposed in [4, 18] by placing a penalty on high-rank filters, namely

$$\min_{\{G_r,\alpha_{rk}\}} \ \frac{1}{2}\sum_{k=1}^{K}\|D_k - \sum_{r=1}^{R}\alpha_{rk}\cdot G_r\|_F^2 + \lambda\sum_{r=1}^{R}\|G_r\|_*, \tag{8}$$

where $\|\cdot\|_*$ is the nuclear norm. [4, 18] highlighted that the choice of λ is a challenging task, and that convergence was slow when estimating high-rank filters. They also propose a second approach based on tensor decomposition [19] that provides faster performance:

$$\min_{\{\alpha_{rk},x_r,y_r\}} \qquad \frac{1}{2}\sum_{k=1}^{K}\|D_k - \sum_{r=1}^{R}\alpha_{rk}\cdot x_r \circ y_r\|_F^2, \tag{9}$$

where x_r and y_r are rank-1 tensors and $\circ$ represents tensor outer product. A reformulation of this problem as a special case of the low-rank basis problem was proposed in [20], but the authors reported that the tensor approach was significantly faster and attained the same accuracy.

An auxiliary variable formulation of (8) given by:

$$\min_{\{G_r,\alpha_{rk},F_r\}} \frac{1}{2}\sum_{k=1}^{K}\|D_k - \sum_{r=1}^{R}\alpha_{rm}G_r\|_F^2 + \frac{\lambda}{2}\sum_{r=1}^{R}\|G_r - F_r\|_F^2$$
$$\text{s.t. rank}(F_r) = 1 \quad \forall r. \tag{10}$$

was proposed in [21] along with an efficient SVD-based generation of the initial solution. The method was shown to be faster than the tensor decomposition approach for small R (< 40) values while attaining comparable accuracy.

3. PROPOSED METHOD

Writing the dictionary update for (4) and coupling the norm constraint with the zero-padding restriction described in Section 2.1 gives the unconstrained problem

$$\min_{\{h_r,v_r\}} \frac{1}{2}\sum_{s=1}^{S}\left\|\sum_{r=1}^{R} v_r * h_r * \mathbf{x}_{r,s} - \mathbf{b}_s\right\|_2^2 + \sum_{r=1}^{R}\iota_{C_{\text{PhN}}}(h_r) + \iota_{C_{\text{PvN}}}(v_r), \tag{11}$$

where $\iota_{C_{\text{PN}}}(\cdot)$ is the indicator function of the constraint set defined in (6), and the zero-padding operator P is applied either along the horizontal or vertical dimension, depending on the filter set.

We approach the solution of (11) by alternating between updating the horizontal filters h_r and the vertical ones v_r. Considering only the solution for the vertical filters v_r (assuming fixed horizontal filters), and reformulating the problem in ADMM-compatible form in a fashion reminiscent of [5] leads to the following expression:

$$\min_{\{v_r,g_r\}} \frac{1}{2}\sum_{s=1}^{S}\left\|\sum_{r=1}^{R} v_r * \mathbf{x}'_{r,s} - \mathbf{b}_s\right\|_2^2 + \sum_{r=1}^{R}\iota_{C_{\text{PvN}}}(g_r)$$
$$s.t. \quad v_r - g_r = 0, \forall r, \tag{12}$$

where $\mathbf{x}'_{r,s}$ is the result of convolving the horizontal filters h_r with the feature maps $\mathbf{x}_{r,s}$. The associated subproblems would be:

$$v_r^{(i+1)} = \arg\min_{v_r} \frac{1}{2}\sum_{s=1}^{S}\left\|\sum_{r=1}^{R} v_r * \mathbf{x}'_{r,s} - \mathbf{b}_s\right\|_2^2 + \frac{\rho}{2}\sum_{r=1}^{R}\left\|v_r - g_r^{(i)} + f_r^{(i)}\right\|_2^2 \qquad (13)$$

$$g_r^{(i+1)} = \arg\min_{g_r} \sum_{r=1}^{R} \iota_{C_{\text{PvN}}}(g_r) + \frac{\rho}{2}\sum_{r=1}^{R}\left\|v_r^{(i+1)} - g_r + f_r^{(i)}\right\|_2^2 \quad (14)$$

$$f_r^{(i+1)} = f_r^{(i)} + v_r^{(i+1)} - g_r^{(i+1)} \qquad (15)$$

Since (14) is of the form:

$$\arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|_2^2 + \iota_{C_{\text{PvN}}}(\mathbf{x}) = \text{prox}_{\iota_{C_{\text{PvN}}}}(\mathbf{y}). \qquad (16)$$

then its minimizer is given by:

$$\text{prox}_{\iota_{C_{\text{PvN}}}}(\mathbf{y}) = \frac{PP^T\mathbf{y}}{\|PP^T\mathbf{y}\|_2}, \qquad (17)$$

For notational simplicity we rewrite (13) as

$$v_r^{(i+1)} = \arg\min_{v_r} \frac{1}{2}\sum_{s=1}^{S}\left\|\sum_{r=1}^{R} v_r * \mathbf{x}'_{r,s} - \mathbf{b}_s\right\|_2^2 + \frac{\rho}{2}\sum_{r=1}^{R}\|v_r - z_r\|_2^2, \qquad (18)$$

where $z_r = g_r^{(i)} - f_r^{(i)}$.

When performing standard CDL [5], the non-separable equivalent of (18) is solved by switching to the DFT domain and solving the associated linear system. In the separable case, however, it's worth noting that since the filters $\{v_r\}$ are 1-D, whereas the coefficient maps $\{\mathbf{x}'_{r,s}\}$ are 2-D, moving directly onto the frequency domain would require the DFT solution ($\hat{v}_r$) to be a 2-D matrix composed of replicating columns. This would mean including an additional constraint and further increasing the complexity of the problem. Instead we choose to rewrite the fidelity ℓ_2-norm term as a sum over columns:

$$\left\|\sum_{r=1}^{R} v_r * \mathbf{x}'_{r,s} - \mathbf{b}_s\right\|_2^2 = \sum_{i=1}^{I}\left\|\sum_{r=1}^{R} v_r * \mathbf{x}'_{r,s}[i] - \mathbf{b}_s[i]\right\|_2^2, \quad (19)$$

where $\mathbf{x}'_{r,s}[i]$ and $\mathbf{b}_s[i]$ are the i-th columns of the corresponding feature map and training image respectively. Replacing the equality in (18):

$$\arg\min_{v_r} \frac{1}{2}\sum_{s=1}^{S}\sum_{i=1}^{I}\left\|\sum_{r=1}^{R} v_r * \mathbf{x}'_{r,s}[i] - \mathbf{b}_s[i]\right\|_2^2 + \frac{\rho}{2}\sum_{r=1}^{R}\|v_r - z_r\|_2^2 \qquad (20)$$

Switching to the DFT domain, and defining $\hat{\mathbf{X}}'_{r,s}[i] = \text{diag}(\hat{\mathbf{x}}'_{r,s}[i])$ gives:

$$\arg\min_{v_r} \frac{1}{2}\sum_{s=1}^{S}\sum_{i=1}^{I}\left\|\sum_{r=1}^{R} \hat{\mathbf{X}}'_{r,s}[i]\hat{v}_r - \hat{\mathbf{b}}_s[i]\right\|_2^2 + \frac{\rho}{2}\sum_{r=1}^{R}\|\hat{v}_r - \hat{z}_r\|_2^2 \qquad (21)$$

Defining:

$$\hat{\mathbf{X}}'_s[i] = (\hat{\mathbf{X}}'_{0,s}[i] \quad \hat{\mathbf{X}}'_{1,s}[i] \quad \ldots) \quad \hat{v} = \begin{bmatrix}\hat{v}_1\\ \hat{v}_2\\ \vdots\\ \hat{v}_R\end{bmatrix} \quad \hat{z} = \begin{bmatrix}\hat{z}_1\\ \hat{z}_2\\ \vdots\\ \hat{z}_R\end{bmatrix} \quad (22)$$

the problem can be expressed as

$$\arg\min_{v_r} \frac{1}{2}\sum_{s=1}^{S}\sum_{i=1}^{I}\left\|\hat{\mathbf{X}}'_s[i]\hat{v} - \hat{\mathbf{b}}_s[i]\right\|_2^2 + \frac{\rho}{2}\|\hat{v} - \hat{z}\|_2^2 \qquad (23)$$

Finally, to overcome the column indexing we define:

$$\hat{\mathbf{X}}'_s = \begin{bmatrix}\hat{\mathbf{X}}'_s[1]\\ \hat{\mathbf{X}}'_s[2]\\ \vdots\\ \hat{\mathbf{X}}'_s[I]\end{bmatrix} \qquad (24)$$

Substituting $\hat{\mathbf{X}}'_s$ and recovering the full vectorized DFT training images $\hat{\mathbf{b}}_s$ leads to the problem being expressed as:

$$\arg\min_{v_r} \frac{1}{2}\sum_{s=1}^{S}\left\|\hat{\mathbf{X}}'_s\hat{v} - \hat{\mathbf{b}}_s\right\|_2^2 + \frac{\rho}{2}\|\hat{v} - \hat{z}\|_2^2, \qquad (25)$$

with solution:

$$\Big(\sum_s \hat{\mathbf{X}}_s'^H\hat{\mathbf{X}}'_s + \rho I\Big)\hat{v} = \sum_s \hat{\mathbf{X}}_s'^H\hat{\mathbf{b}}_s + \rho\hat{z} \qquad (26)$$

Due to the commutativity property of the convolution operation, the update for the horizontal filters h_r can be easily derived by fixing the vertical filters, defining $\mathbf{x}'_{r,s} = v_r * \mathbf{x}_{r,s}$, and following an analogous chain of derivations as the one described in this section.

3.1. Implementation remarks

The linear system given by Eq. (26) is solved by applying Conjugate Gradient (CG). Furthermore, in order to minimize the number of inner CG iterations, we use the solution for each previous update as the initial value, as suggested in [6].

The full dictionary learning algorithm is implemented by combining the proposed update method for $\{v_r\}$ and $\{h_r\}$ with the ADMM-based sparse coding update proposed in [5]. Based on standard non-separable implementations, and the results provided by [15], we interleave a single iteration of each update per outer loop, and transfer the auxiliary variables of each ADMM framework across the other update steps, which has been shown to provide the most stable convergence ratio among the other possible choices.

4. RESULTS

In this section we assess the performance of the proposed separable dictionary learning method in terms of reconstruction performance for a denoising CSC task, along with convergence and computational runtime for the learning process.

4.1. Experimental framework

For the denoising comparisons, we used a set of 5 well-known images corrupted with AWGN ($\sigma = 0.2$), to perform CBPDN using the following labeled set of filters of size 12×12:

Table 1: Denoising performance (SSIM)

	barbara	mandrill	parrots	boats	goldhill	Time (sec)
nat-sep	0.637	0.5248	0.7219	0.6532	0.673	60.57
non-sep	0.633	0.530	0.7225	0.6507	0.6763	104.74
apr-sep	0.6122	0.5186	0.7132	0.6396	0.664	60.64

- **Nat-sep**: 36 Natively learned separable filters (our proposed method)
- **Apr-sep**: 36 Separable filters approximated from 36 non-separable ones via [21]
- **Non-sep**: 36 Standard non-separable filters learned via [5]

Since the CBPDN problem (as defined in Eq. (1)) has a tunable parameter λ, in order to ensure fair evaluation we solve for a grid of λ values, and compare only the optimal performance for each of the evaluated filter sets. An example of the entire simulation results for a single image is given in Figure 1.

For the separable dictionaries (nat-sep and apr-sep), we use the ℓ_1 version of the FISTA-based CBPDN solver proposed in [22] that exploits filter separability by computing the convolutions in the spatial domain. For the non-separable dictionaries (non-sep), we use the ADMM-based solver from [5], which is considered to be state-of-the-art for this problem.

For the computational performance simulations, we evaluate the learning time for different training set sizes (S) and filter set sizes ($R = K$) against a state-of-the-art ADMMM-based non-separable CDL method [5]. These simulations were performed on an Intel Xeon E5-2640 CPU (2,50 GHz , 128Gb RAM, 2x NVidia Tesla K40m GPU). Our matlab code [23] can be used to reproduce our experimental results.

4.2. Experiments

In Table 1 we illustrate the results of the denoising comparisons between the 3 evaluated filter sets in terms of SSIM metric, and report the average runtime for each method across the grid of λ values. It can be observed that the natively separable filters consistently outperform the approximated (separable) ones, and show equivalent performance to the non-separable filters. The runtime results also show that performing CSC with separable filters is almost two times faster than doing it with non separable ones, which is consistent with the results reported in [22]. We also show in Figure 1 the entire set of denoising simulations across the λ grid for a single image.

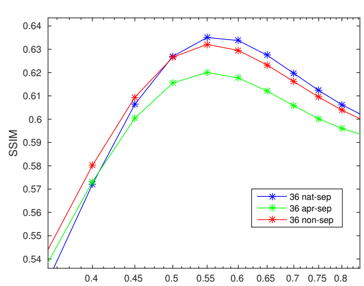
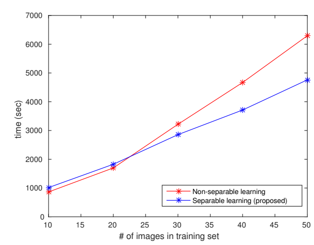
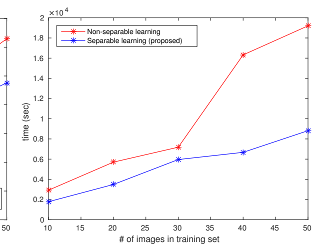


Fig. 1: Denoising results on λ grid for 'barbara' image, where *apr-sep, nat-sep* and *non-sep* are the labels defined in Section 4.1

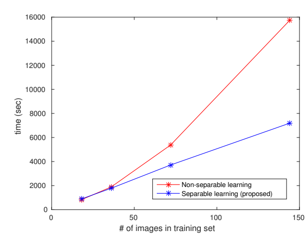
We report in Figure 2 (a) and (b) the computational performance comparisons in the learning process for 2 different CG tolerance values, in terms of runtime (seconds) vs image training set size. We consider a fixed number of 36 separable and non-separable filters for this simulation, and measure the runtime for both training methods for a fixed number of iterations (200). As can be observed in the graph, when the CG tolerance is 10^{-3} the proposed separable method is slightly slower than its non-separable counterpart for small values of S, and outperforms it when S increases. When the tolerance value is 10^{-5}, the proposed method significantly outperforms the other one as S increases. Figure 2 (c) depicts a similar runtime comparison where the training set size is fixed ($S = 20$) and the dictionary size (number of filters) is varied (the tolerance value used is 10^{-3}). In this case it is also clear that the proposed method is substantially faster than the non-separable method as the number of filters increases.



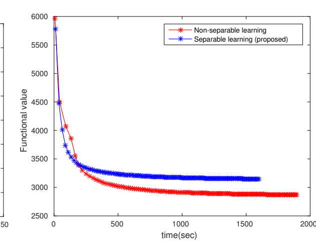
(a) Tol=10^{-3}



(b) Tol=10^{-5}



(c) Time vs filter set size



(d) Functional value example

Fig. 2

An example of the functional value behaviour for a training set size of $S = 20$ is shown in Figure 2 (d) for 200 iterations. It can be seen from the graph that the proposed separable method converges to a slightly higher functional value than the non-separable method. However, this difference does not seem to have a significant impact on the performance quality of the learned separable filters, as can be seen on Table 1.

5. CONCLUSIONS

We have proposed an efficient method to learn separable dictionary filters directly from an image training set, without the need to previously compute a set of non-separable filters. Our results show that the separable filters learned through this method, when evaluated through a CSC denoising task, consistently outperform approximated separable filters, and attain the same reconstruction quality as when using standard non-separable filters. Furthermore, the proposed separable learning method is substantially faster than its state-of-the-art non-separable counterpart when either the training set or the number of filters to estimate is large.

6. REFERENCES

[1] Julien Mairal, Francis Bach, and Jean Ponce, "Sparse modeling for image and vision processing," *Foundations and Trends in Computer Graphics and Vision*, vol. 8, no. 2-3, pp. 85–283, 2014.

[2] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 2528–2535.

[3] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Speeding up convolutional neural networks with low rank expansions," in *Proceedings of the British Machine Vision Conference*. 2014, BMVA Press.

[4] R. Rigamonti, A. Sironi, V. Lepetit, and P. Fua, "Learning separable filters," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, June 2013, pp. 2754–2761.

[5] Brendt Wohlberg, "Efficient algorithms for convolutional sparse representations," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 301–315, Jan. 2016.

[6] C. Garcia-Cardona and B. Wohlberg, "Convolutional Dictionary Learning," *ArXiv e-prints*, Sept. 2017.

[7] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*, 1999, vol. 5, pp. 2443–2446 vol.5.

[8] M. Mrup and M. N. Schmidt, "Transformation invariant sparse coding," in *2011 IEEE International Workshop on Machine Learning for Signal Processing*, Sept 2011, pp. 1–6.

[9] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *2011 International Conference on Computer Vision*, Nov 2011, pp. 2018–2025.

[10] Q. Barthelemy, A. Larue, A. Mayoue, D. Mercier, and J. I. Mars, "Shift amp; 2d rotation invariant sparse coding for multivariate signals," *IEEE Transactions on Signal Processing*, vol. 60, no. 4, pp. 1597–1611, April 2012.

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

[12] H. Bristow, A. Eriksson, and S. Lucey, "Fast convolutional sparse coding," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, June 2013, pp. 391–398.

[13] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[14] Michal Ŝorel and Filip Ŝroubek, "Fast convolutional sparse coding using matrix inversion lemma," *Digital Signal Processing*, vol. 55, pp. 44 – 51, 2016.

[15] Cristina Garcia-Cardona and Brendt Wohlberg, "Subproblem coupling in convolutional dictionary learning," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, Beijing, China, Sept. 2017.

[16] K. Degraux, U. S. Kamilov, P. T. Boufounos, and D. Liu, "Online Convolutional Dictionary Learning for Multimodal Imaging," *ArXiv e-prints*, June 2017.

[17] J. Liu, C. Garcia-Cardona, B. Wohlberg, and W. Yin, "Online Convolutional Dictionary Learning," *ArXiv e-prints*, June 2017.

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

[19] Tamara G. Kolda and Brett W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[20] Yuji Nakatsukasa, Tasuku Soma, and André Uschmajew, "Finding a low-rank basis in a matrix subspace," *Mathematical Programming*, vol. 162, no. 1, pp. 325–361, Mar 2017.

[21] P. Rodrguez, "Alternating optimization low-rank expansion algorithm to estimate a linear combination of separable filters to approximate 2d filter banks," in *2016 50th Asilomar Conference on Signals, Systems and Computers*, Nov 2016, pp. 954–958.

[22] G. Silva, J. Quesada, P. Rodrguez, and B. Wohlberg, "Fast convolutional sparse coding with separable filters," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 6035–6039.

[23] J. Quesada and P. Rodriguez, "Separable filter learning," Available at `https://sites.google.com/pucp.pe/jquesada`.