COMBINATORIAL SEPARABLE CONVOLUTIONAL DICTIONARIES

Jorge Quesada† *Gustavo Silva*† *Paul Rodriguez*† *Brendt Wohlberg**

†Department of Electrical Engineering, Pontificia Universidad Catolica del Peru, Lima, Peru
*T-5, Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM, USA

ABSTRACT

Recent works have considered the use of a linear combination of separable filters to approximate a non-separable filter bank (FB) to obtain computational advantages in CNNs and convolutional sparse representations / coding (CSR / CSC). However, it has been recently shown that there are advantages to directly solving the convolutional dictionary learning (CDL) problem considering a separable FB.

A separable filter bank of M 2-d filters is typically constructed from a paired set of M horizontal filters and M vertical filters. In contrast, here we propose an outer product construction involving all possible combinations of vertical and horizontal filters, so that M vertical and M horizontal filters generate M^2 2-d filters. Our computational experiments show that this alternative form results in a reduction in computation time of 10% and 80% for the CDL and CSC problems respectively, while matching the reconstruction performance of the typical separable FB approach for the same cardinality.

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 ability to represent entire images, as opposed to their patch-based counterparts [2]. The most common form of Convolutional Sparse Coding (CSC) problem is Convolutional Basis Pursuit Denoising (CBPDN)

$$\underset{\{\mathbf{x}_m\}}{\arg\min} \frac{1}{2}\Big\| \sum_m \mathbf{d}_m * \mathbf{x}_m - \mathbf{s}\Big\|_2^2 + \lambda \sum_m \Big\|\mathbf{x}_m\Big\|_1 , \quad (1)$$

where $\mathbf{s}$ is the observed image, $\{\mathbf{x}_k\}$ is the coefficient map set, and $\{\mathbf{d}_k\}$ are the non-separable dictionary filters. The corresponding Convolutional Dictionary Learning (CDL) problem is

$$\underset{\{\mathbf{x}_{m,k}\}\{\mathbf{d}_m\}}{\arg\min} \frac{1}{2}\sum_k \Big\| \sum_m \mathbf{d}_m * \mathbf{x}_{m,k} - \mathbf{s}_k\Big\|_2^2 + \lambda \sum_k \sum_m \Big\|\mathbf{x}_{m,k}\Big\|_1 \quad \text{s.t.} \quad \|\mathbf{d}_m\|_2 = 1 \quad \forall m , \quad (2)$$

where $\{\mathbf{s}_k\}$ is a set of training images, and the constraint on the filter norms is used to avoid scaling ambiguities. It has been shown that using separable filters in Convolutional Neural Network (CNN) applications [3, 4] and as dictionaries in CSC [5] can provide significant improvements in computational performance with respect to a non-separable implementation, with little loss in accuracy or reconstruction quality. As a consequence, some works have proposed learning separable filter banks directly from an image training set [6, 7], by solving a separable version of the CDL problem,

$$\underset{\{\mathbf{x}_{m,k}\}\{\mathbf{v}_m\}\{\mathbf{h}_m\}}{\arg\min} \frac{1}{2}\sum_k \Big\| \sum_m \mathbf{v}_m * \mathbf{h}_m * \mathbf{x}_{m,k} - \mathbf{s}_k\Big\|_2^2 + \lambda \sum_k \sum_m \Big\|\mathbf{x}_{m,k}\Big\|_1 \quad \text{s.t.} \quad \|\mathbf{v}_m\|_2 = \|\mathbf{h}_m\|_2 = 1 \;\forall m , \quad (3)$$

where $\{\mathbf{h}_m\}$ and $\{\mathbf{v}_m\}$ are the horizontal and vertical components of each filter. Natively separable filters have been shown to outperform the previous approach via separable approximation from a non-separable set of filters [6].

Existing 2-d separable filter banks are constructed by pairing a set of M vertical and M horizontal filters, the directional filters in each pair being convolved to give a single 2-d filter, with the resulting filter bank consisting of M 2-d filters. In contrast, in this paper we explore an alternative construction in which the directional filters are not paired, so that the resulting 2-d filter bank consists of the M^2 2-d filters composes of all combinations of a vertical and a horizontal filter. The corresponding CDL problem can be posed as

$$\underset{\{\mathbf{x}_{m,n,k}\}\{\mathbf{v}_m\}\{\mathbf{h}_n\}}{\arg\min} \frac{1}{2}\sum_k \Big\| \sum_{m,n} \mathbf{v}_m * \mathbf{h}_n * \mathbf{x}_{m,n,k} - \mathbf{s}_k\Big\|_2^2 + \lambda \sum_k \sum_m \sum_n \Big\|\mathbf{x}_{m,n,k}\Big\|_1 \quad \text{s.t.} \quad \|\mathbf{v}_m\|_2 = \|\mathbf{h}_n\|_2 = 1 \quad \forall m, \forall n . \quad (4)$$

The proposed method is derived as a natural extension of existing separable CDL algorithms [6, 7], and compared against a baseline separable CDL method. The computational results in Section 4 show that the dictionary filters produced by this method attain speedups both in learning and reconstruction runtime, with an equivalent denoising performance with respect to the baseline.

2. PREVIOUS RELATED WORK

2.1. Non-separable (standard) dictionary learning

The CDL problem (2) is non-convex when simultaneously minimizing for both variables, $\{\mathbf{x}_{m,k}\}$ and $\{\mathbf{d}_m\}$, but is convex when either of them is kept constant. Therefore, the most common optimization approach consists of alternating between the updates for the feature maps $\{\mathbf{x}_{m,k}\}$ (sparse coding) and the filters $\{\mathbf{d}_m\}$ (dictionary update). This section will address the main existing dictionary update methods [8, 9], which correspond to the solution of a constrained convolutional form [2, 9] of the Method of Optimal Directions (MOD) [10]

$$\begin{aligned}\underset{\{\mathbf{d}_m\}}{\arg\min}\ \frac{1}{2}\sum_k \Big\| \sum_m \mathbf{d}_m * \mathbf{x}_{m,k} - \mathbf{s}_k \Big\|_2^2 \\ \text{s.t. } \ \|\mathbf{d}_m\|_2 = 1 \ \ \forall m \ ,\end{aligned} \tag{5}$$

where $\{\mathbf{x}_{m,k}\}$ is a given coefficient map set.

Early methods solved this problem in the spatial domain, [11, 12, 13] while more recent approaches tackle the most computationally demanding components of the problem in the frequency domain [14, 2, 9]. In the latter case, it is important to give an adequate spatial support to the target filters through zero-padding, which is usually denoted by a zero-padding projection operator P, and combined with the normalization constraint in the constraint set

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

This allows the dictionary update to be written in unconstrained form

$$\underset{\{\mathbf{d}_m\}}{\arg\min}\ \frac{1}{2}\sum_k \Big\| \sum_m \mathbf{d}_m * \mathbf{x}_{m,k} - \mathbf{s}_k \Big\|_2^2 + \sum_m \iota_{C_{\text{PN}}}(\mathbf{d}_m) \ , \tag{7}$$

where $\iota_{C_{\text{PN}}}(\cdot)$ is the indicator function of the constraint set C_{PN}. Most approaches to solving (7) are based on the Alternating Direction Method of Multipliers (ADMM) [15] framework, but the advantages of the Accelerated Proximal Gradient (APG) framework [16] for this problem have recently been noted [17, 18, 7, 9, 19].

2.2. Separable filters for CSR

The idea of using separable filters in order to improve computational runtime was initially introduced in the context of CNNs [20], and was approached by estimating a separable set $\{\mathbf{g}_r\}$ from a given larger set of non-separable filters $\{\mathbf{d}_m\}$, by placing a penalty on the rank of the filters, which can be expressed as

$$\underset{\{\mathbf{g}_r\}\{\alpha_{r,m}\}}{\arg\min}\ \frac{1}{2}\sum_m \Big\|\mathbf{d}_m - \sum_r \alpha_{r,m}\cdot \mathbf{g}_r\Big\|_F^2 + \lambda \sum_r \Big\|\mathbf{g}_r\Big\|_* \ , \tag{8}$$

where $\|\cdot\|_*$ is the nuclear norm. Since this method has suboptimal convergence properties [20, 3], a faster approach based on the Canonical Polyadic Decomposition has been proposed [21]

$$\underset{\{\alpha_{r,m}\}\{\mathbf{h}_r\}\{\mathbf{v}_r\}}{\arg\min}\ \frac{1}{2}\sum_m \Big\|\mathbf{d}_m - \sum_r \alpha_{r,m}\cdot \mathbf{h}_r \circ \mathbf{v}_r\Big\|_F^2 \ , \tag{9}$$

where $\mathbf{h}_r$ and $\mathbf{v}_r$ are rank-1 tensors and $\circ$ represents tensor outer product. A more computationally efficient reformulation of (8), given by

$$\begin{aligned}\underset{\{\mathbf{g}_r\}\{\alpha_{r,m}\}\{\mathbf{f}_r\}}{\arg\min}\ \frac{1}{2}\sum_m \Big\|\mathbf{d}_m - \sum_r \alpha_{r,m}\cdot \mathbf{g}_r\Big\|_F^2 \\ + \lambda \sum_r \Big\|\mathbf{g}_r - \mathbf{f}_r\Big\|_F^2 \quad \text{s.t.} \quad \text{rank}(\mathbf{f}_r) = 1 \quad \forall r\end{aligned} \tag{10}$$

was proposed in [22], 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, and evaluated in a CSC setting in [5].

More recently, [6] proposed to learn the separable filter set natively by solving and showed that the resulting dictionary filters entailed equivalent performance to their non-separable counterparts learned through an analogous method. A more computationally efficient variant based on the APG framework was proposed in [7], where the separability was enforced through a rank-1 constraint.

2.3. Rank-1 Accelerated Proximal Gradient

Learning separable filters, such as vertical and horizontal filters, is analogous to learning rank-one 2-d filters. Motivated by this observation, [7] applied an additional constraint on the typical non-separable formulation (5) to enforce that the obtained dictionaries be rank-one.

$$\begin{aligned}\underset{\{\mathbf{d}_m\}}{\arg\min}\ \frac{1}{2}\sum_k \Big\| \sum_m \mathbf{d}_m * \mathbf{x}_{m,k} - \mathbf{s}_k \Big\|_2^2 + \sum_m \iota_{C_{\text{PR}}}(\mathbf{d}_m) \\ \text{s.t.} \quad \|\mathbf{d}\|_2 = 1,\end{aligned} \tag{11}$$

where $C_{\text{PR}} = \{\mathbf{z} \in \mathbb{R}^N : (I - PP^T)\mathbf{z} = 0, \ \text{rank}(\mathbf{z}) = 1\}$. The APG solution for this problem was found to be more efficient than the ADMM solution [7]. The APG formulation is given by

$$\mathbf{d_m}^{(i+1)} = \text{Prox}_{\iota_{C_{\text{PR}}}}\Big(\mathbf{d_m}^{(i)} - \frac{1}{L}\nabla F(\mathbf{d}_m^{(i)})\Big) \ , \tag{12}$$

where the proximal operator of the constraint set is given by

$$[U\Sigma V^T] = SVD(PP^T\mathbf{y}) \tag{13}$$

$$\text{prox}_{\iota_{C_{\text{PR}}}}(\mathbf{y}) = U\Sigma_0 V^T \ , \tag{14}$$

where Σ_0 is obtained from Σ by zeroing all but the first singular value, i.e. if $\Sigma = \text{diag}(\sigma_0, \sigma_1, \ldots)$ then $\Sigma_0 = \text{diag}(\sigma_0, 0, \ldots, 0)$.

3. PROPOSED METHOD

Formulating the dictionary learning update for (4) and integrating the norm constraint and the zero padding operator into a constraint set as described in Section 2.1 results in the unconstrained problem

$$\begin{aligned}\underset{\{\mathbf{v}_m\}\{\mathbf{h}_n\}}{\arg\min}\ \frac{1}{2}\sum_k \Big\| \sum_{m,n} \mathbf{v}_m * \mathbf{h}_n * \mathbf{x}_{m,n,k} - \mathbf{s}_k \Big\|_2^2 \\ + \sum_m \iota_{C_{\text{P}_\text{v}\text{N}}}(\mathbf{v}_m) + \sum_n \iota_{C_{\text{P}_\text{h}\text{N}}}(\mathbf{h}_n)\end{aligned} \tag{15}$$

where $\iota_{C_{\text{P}_\text{h}\text{N}}}(\cdot)$ and $\iota_{C_{\text{P}_\text{v}\text{N}}}(\cdot)$ are the indicator functions of the constraint sets $C_{\text{P}_\text{h}\text{N}}$ and $C_{\text{P}_\text{v}\text{N}}$ (analogous to (6)), with zero-padding operators applied along the horizontal and vertical dimensions, respectively.

In a similar fashion to [6], the solution of (15) is approached by alternating between updates of the vertical and horizontal filters. When considering only the updates for the vertical filters, the problem can be simplified by integrating the fixed horizontal filters and the feature maps into an *auxiliary* feature map variable $\mathbf{x}_{\mathbf{v}m,k}$:

$$\mathbf{x}_{\mathbf{v}m,k} = \sum_n \mathbf{h}_n * \mathbf{x}_{m,n,k}, \tag{16}$$

which leads to the problem

$$\underset{\{\mathbf{v}_m\}}{\arg\min}\ \frac{1}{2}\sum_k \Big\| \sum_m \mathbf{v}_m * \mathbf{x}_{\mathbf{v}m,k} - \mathbf{s}_k \Big\|_2^2 + \sum_{m_1} \iota_{C_{\text{P}_\text{v}\text{N}}}(\mathbf{v}_{m_1}) \ . \tag{17}$$

For simplicity of notation, we define a linear operator $\mathbf{X}_{\text{v},m,k}$ such that

$$\mathbf{X}_{\text{v},m,k}\mathbf{v}_m = \mathbf{x}_{\text{v},m,k} * \mathbf{v}_m$$

, which results in the formulation

$$\arg\min_{\mathbf{V}} \frac{1}{2} \sum_k \left\| \mathbf{X}_{\text{v},k} \mathbf{V} - \mathbf{s}_k \right\|_2^2 + \iota_{C_{\text{P}_\text{v}\text{N}}}(\mathbf{V}) \ , \tag{18}$$

where $\mathbf{X}_{\text{v},k} = [\mathbf{X}_{\text{v},1,k}, \cdots, \mathbf{X}_{\text{v},M,k}]$ and $\mathbf{V} = [\mathbf{v}, \cdots, \mathbf{v}_M]^T$. The problem can be further simplified to

$$\arg\min_{\mathbf{V}} \frac{1}{2} \left\| \mathbf{X}_{\text{v}} \mathbf{V} - \mathbf{S} \right\|_2^2 + \iota_{C_{\text{P}_\text{v}\text{N}}}(\mathbf{V}) \tag{19}$$

where $\mathbf{X}_{\text{v}} = [\mathbf{X}_{\text{v},1}, \cdots, \mathbf{X}_{\text{v},K}]^T$ and $\mathbf{S} = [\mathbf{s}_1, \cdots, \mathbf{s}_K]^T$. Problem (18) can then be solved through the APG-based method derived in [18], with update iterations are given by

$$\nabla F(\hat{\mathbf{V}}^{(i)}) = \hat{\mathbf{X}}_{\text{v}}^H (\hat{\mathbf{X}}_{\text{v}} \hat{\mathbf{V}}^{(i)} - \hat{\mathbf{S}}) \tag{20}$$

$$\alpha^{(i)} = ||\nabla F(\hat{\mathbf{V}}^{(i)})||_2^2 / ||\hat{\mathbf{X}}_{\text{v}}^H \nabla F(\hat{\mathbf{V}}^{(i)})||_2^2 \tag{21}$$

$$\mathbf{V}^{(i+1)} = \text{prox}_{\iota_{C_{\text{P}_\text{v}\text{N}}}} (\mathbf{V}^{(i)} - \alpha^{(i)} \nabla F(\mathbf{V}^{(i)})) \ , \tag{22}$$

where the proximal operator for the third update is given by

$$\text{prox}_{\iota_{C_{\text{P}_\text{v}\text{N}}}}(\mathbf{y}) = \frac{PP^T \mathbf{y}}{||PP^T \mathbf{y}||_2} \ .$$

Due to the commutative property of the convolution operation, a similar approach can be followed to derive derive the updates for the horizontal filters, by simply composing the auxiliary feature maps as $\mathbf{x}_{\mathbf{v}n,k} = \sum_m \mathbf{v}_m * \mathbf{x}_{m,n,k}$.

4. EXPERIMENTAL RESULTS

4.1. Experimental Setup

Our experimental setup is similar to that in [6] and [7]. Two distinct set of experiments were carried out on a desktop computer equipped with an Intel i7-7700K CPU (4.20 GHz, 8MB Cache, 32GB RAM): (i) *learning* simulations, in order to assess the convergence and runtime performance of the proposed method, and (ii) *denoising* simulations, in order to analyze the reconstruction quality of the attained filter banks.

The denoising comparisons were performed on a set of standard images corrupted with Additive White Gaussian Noise (AWGN) with $\sigma = 0.2$, using filter banks with varying cardinalities and filters of size 24×24 (determined in [23] to be the optimal size) learned through one of the two following methods:

- **Pair-SepF**: baseline method. A set of M^2 natively separable filters learned through the APG-based method proposed in [7] (to the best of our knowledge, the most computationally efficient separable CDL method to date).
- **Comb-SepF**: proposed method. $M \times M$ separable filter bank composed of of M vertical and M horizontal separable filters learned through the approach described in Section 3.

The denoising comparisons were performed using the CBPDN solver proposed in [5], which is optimized to exploit filter separability, and evaluated on a grid of λ values, from which we report only the peak performance for each case. It is also worth mentioning that when using the filters obtained by **Comb-SepF**, the solver is further optimized in order to exploit the combinatorial nature of the filter bank and thus improve the sparse coding runtime (see Figure 2 (b)), by applying the simplification shown in Eq. (16). We also consider the L0 % sparsity measure attained in each case, defined as $100 \cdot ||x||_0/N$, where x is the feature map set and N is the number of pixels in a given image.

The learning comparisons assess the computational performance in terms of learning runtime of our proposed method with respect to the baseline [7] as well as the functional value behaviour, for a fixed value of 500 iterations.

4.2. Learning runtime simulations

In Figure 1 we report the computational performance and functional behavior comparison in logarithmic time for the evaluated methods when trained with a batch of 50 training images, when learning target filter banks of sizes [36, 144] and $[6 \times 6, 12 \times 12]$ for the **Pair-SepF** and **Comb-SepF** approaches, respectively. As can be observed, both methods converge to roughly similar functional values (the proposed method converges to a slightly higher value, which can be explained by its more restricted solution space), with the proposed method attaining from 6% to 10% speedup with respect to the baseline.

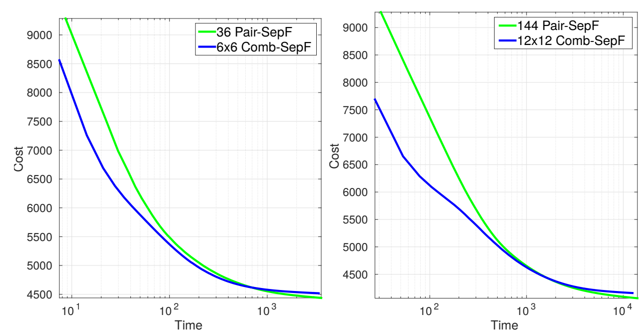


Fig. 1: Functional value behaviour when learning with a batch of 50 images

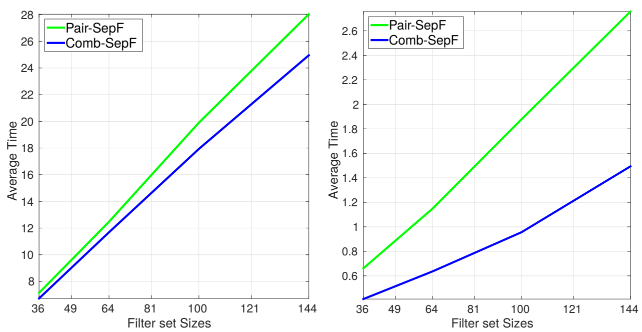
The learning runtime comparison of both methods can be observed in more detail in Figure 2 (a). As can be seen from the graph, the proposed method attains a speedup ranging from 6% to 10%, and increasing as the filter bank size gets larger.

4.3. Denoising restoration simulations

In Table 1 we report the results for the denoising comparisons among the evaluated methods in terms of the PSNR metric, as well as the attained L0% sparsity measure in each case. It can be observed from the table that the filters obtained through our proposed method show equivalent performance with respect to the baseline filters, the difference between them being negligible in the two considered metrics (as was previously mentioned, only the peak performance for each image over the λ grid is reported). It is also worth noting that, as was reported in [23], the best performance is not always achieved by the largest filter bank, but rather a particular cardinality for each given image.

Table 1: Denoising via CBPDN for different filter bank cardinalities in images corrupted with AWGN level $\sigma = 0.2$.

	Barbara		**Mandrill**		**Peppers**		**Boats**		**House**		**GoldHill**		**Monarch**		**Airplane**	
	PSNR	L0 %	PSNR	L0 %	PSNR	L0 %	PSNR	L0 %	PSNR	L0 %	PSNR	L0 %	PSNR	L0 %	PSNR	L0 %
36 **Pair-SepF**	23.24	5.42	20.95	7.93	25.00	1.84	24.18	2.77	24.97	3.25	24.78	1.14	24.34	3.19	24.12	3.96
6×6 **Comb-SepF**	23.22	5.93	20.93	8.59	24.92	1.50	24.05	3.04	24.74	3.65	24.71	1.23	24.26	3.37	24.00	3.59
64 **Pair-SepF**	23.38	4.49	20.95	10.17	24.97	1.98	24.15	2.26	24.91	2.75	24.77	0.88	24.30	3.58	24.10	3.22
8×8 **Comb-SepF**	23.33	4.79	20.92	10.67	24.92	2.10	24.03	2.39	24.72	2.91	24.71	0.89	24.20	2.69	24.01	3.37
100 **Pair-SepF**	23.47	5.47	20.94	8.14	24.98	1.61	24.15	2.81	24.92	3.33	24.79	1.13	24.31	3.18	24.08	3.94
10×10 **Comb-SepF**	23.36	5.59	20.93	8.21	24.91	2.07	24.06	2.81	24.83	3.39	24.74	1.10	24.23	3.17	24.01	3.94
144 **Pair-SepF**	23.45	3.95	20.97	9.15	24.97	1.66	24.11	3.31	24.91	2.36	24.76	1.37	24.29	3.69	24.06	2.77
12×12 **Comb-SepF**	23.33	5.47	20.94	9.25	24.90	1.68	24.01	3.34	24.77	2.47	24.74	1.32	24.24	3.71	24.00	2.81



(a) CDL (learning)

(b) CSC (denoising)

Fig. 2: Comparison of mean time per iteration

One can also observe from Figure 2 (b) that after modifying [5] to exploit the combinatorial structure of the proposed filter bank, a speedup between 60% and 80% with respect to ”standard” separable filters is attained in the denoising process. As in the learning case, the denoising speedup also increases as the filter bank cardinality grows larger. An additional contribution of the proposed method is that the filter bank obtained by this approach significantly reduces the amount of memory required to store and subsequently use the filters, by a factor of M, where M^2 is the cardinality of the original filter bank. This is clearly a useful property when dealing with memory-constrained tasks.

5. CONCLUSION

We have proposed and evaluated a novel formulation to solve the separable Convolutional Dictionary Learning (CDL) problem by constructing a separable filter bank as the result from all posible combinations of outer products between a set of M vertical and horizontal filters. Our numerical results show that the filters learned through our proposed method consistently attain indistinguishable restoration performance with respect to a baseline method with the same filter size and cardinality.

Furthermore, the proposed method delivers a computational runtime improvement in both learning and sparse coding contexts, the highest speedup ratios being attained when the filter banks cardinalities are large. An additional contribution of our method is that the learned filters significantly reduce the memory representation space required to use them.

6. REFERENCES

[1] J. Mairal, F. Bach, and J. 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] B. Wohlberg, “Efficient algorithms for convolutional sparse representations,” *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 301–315, Jan. 2016.

[3] 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.

[4] 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.

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

[6] J. Quesada, P. Rodriguez, and B. Wohlberg, “Separable dictionary learning for convolutional sparse coding via split updates,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4094–4098.

[7] G. Silva, J. Quesada, and P. Rodríguez, “Efficient separable filter estimation using rank-1 convolutional dictionary learning,” in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, Sept. 2018.

[8] C. Garcia-Cardona and B. Wohlberg, “Subproblem coupling in convolutional dictionary learning,” in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, Beijing, China, Sept. 2017, pp. 1697–1701.

[9] C. Garcia-Cardona and B. Wohlberg, “Convolutional dictionary learning: A comparative review and new algorithms,” *IEEE Transactions on Computational Imaging*, vol. 4, no. 3, pp. 366–381, Sept. 2018.

[10] K. Engan, S. O. Aase, and J. Hakon Husoy, “Method of optimal directions for frame design,” in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1999, vol. 5, pp. 2443–2446.

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

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

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

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

[15] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. 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.

[16] N. Parikh, S. Boyd, et al., "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[17] G. Silva and P. Rodríguez, "Efficient algorithm for convolutional dictionary learning via accelerated proximal gradient consensus," in *IEEE International Conference on Image Processing (ICIP)*, Athens, Greece, Oct. 2018.

[18] G. Silva and P. Rodriguez, "Efficient convolutional dictionary learning using partial update fast iterative shrinkage-thresholding algorithm," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Calgary, Canada, 2018.

[19] C. Garcia-Cardona and B. Wohlberg, "Convolutional dictionary learning for multi-channel signals," in *Proceedings of the 2018 Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, USA, Oct. 2018.

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

[21] T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

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

[23] G. Silva, J. Quesada, and P. Rodriguez, "Hyper-parameter selection on convolutional dictionary learning," Submitted to *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*,Brighton, UK, May 2019.