SUBPROBLEM COUPLING IN CONVOLUTIONAL DICTIONARY LEARNING

Cristina Garcia-Cardona

CCS-3, CCS Division
Los Alamos National Laboratory
Los Alamos, NM 87545, USA

Brendt Wohlberg

T-5, Theoretical Division
Los Alamos National Laboratory
Los Alamos, NM 87545, USA

ABSTRACT

The current leading algorithms for both convolutional sparse coding and dictionary learning are based on variable splitting and Augmented Lagrangian methods. The dictionary learning algorithms alternate between sparse coding and dictionary subproblems, typically interleaving the updates for each of these two subproblems. Due to the variable splitting, in each subproblem one of these two variables must be chosen to be passed to the other subproblem. We perform a careful comparison of the algorithm convergence resulting from the different choices in conjunction with a number of different algorithms for the dictionary subproblem, showing that one of these choices consistently provides the best convergence.

***Index Terms*—** Convolutional Sparse Representations, Convolutional Dictionary Learning, ADMM

1. INTRODUCTION

Convolutional sparse representations differ from standard sparse representations in that they represent an entire signal in terms of sums over a set of convolutions with dictionary filters [1]. There is an efficient algorithm for the convolution sparse coding problem [2], but the corresponding dictionary learning problem is more difficult. A number of different approaches have been proposed [3, 4, 5, 6], but a comprehensive comparison has yet to be performed. All of these methods follow the same general structure, alternating between solving a sparse code subproblem and a dictionary subproblem. The sparse coding subproblem is defined as a convolutional variant of the Basis Pursuit Denoising (BPDN) problem, referred to as Convolutional BPDN (CBPDN),

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

and the dictionary subproblem[1] is a convolutional variant of the Method of Optimal Directions (MOD) [7], which will be referred to here as Convolutional MOD (CMOD) [4],

$$\mathop{\arg\min}_{\{\mathbf{d}_m \in C_{\mathrm{PN}}\}} \; \frac{1}{2}\sum_k \Big\|\sum_m \mathbf{d}_m * \mathbf{x}_{k,m} - \mathbf{s}_k\Big\|_2^2 \,, \qquad (2)$$

where $\{\mathbf{d}_m\}$ represents a set of M dictionary filters, $\{\mathbf{x}_{k,m}\}$ a set of M coefficient maps for each of the K signals, $\mathbf{s}_k$, each of which has N pixels, $*$ denotes the convolution operator and C_{PN} is constraint set that fixes the spatial support and scale of the filters [4].

This research was supported by the U.S. Department of Energy via the LANL/LDRD Program.

[1]Usually referred to as the *dictionary update*, but to avoid confusion with the ADMM updates within the subproblems, here we will refer to it as the *dictionary subproblem*.

The most efficient solution for subproblem (1) is an iterative algorithm [2] based on the Alternating Direction Method of Multipliers (ADMM) [8], and most solutions for subproblem (2) are also ADMM algorithms [3, 4, 5, 6]. In both cases the subproblems are addressed within the ADMM framework via variable splitting, introducing an auxiliary variable that is constrained to be equal to the primary solution variable in the original formulation; e.g. in (1) an auxiliary variable $\{\mathbf{y}_{k,m}\}$ is introduced, constrained to be equal to primary variable $\{\mathbf{x}_{k,m}\}$. The usual approach to combining these iterative algorithms for the two subproblems is to interleave their updates, taking a few passes over the updates of the one subproblem before switching to the other subproblem.

When switching between subproblems, a natural question that arises is which of the primary and auxiliary variables in the one subproblem should be transferred to the other; e.g. after a pass over the sparse coding subproblem, should $\{\mathbf{x}_{k,m}\}$ or $\{\mathbf{y}_{k,m}\}$ be passed to the dictionary learning subproblem as the working coefficient maps. This important question has received little attention other than in [4, 9]: the original interleaved algorithm [3] used the primary variables as a natual consequence of the derivation of the entire dictionary learning algorithm within the Augmented Lagrangian framework, and this structure was largely retained in later work by other authors [5, 6].

The purpose of the present paper is to perform a thorough comparison of the different coupling options for the current leading dictionary solving algorithms. We show that the choice of primary or auxiliary coefficient map variable is more critical than the choice for the dictionary variable, and that the use of the auxiliary variable for both subproblems consistently provides faster and more reliable convergence.

2. DICTIONARY SUBPROBLEM ALGORITHMS

We concentrate on the dictionary subproblem (2) since the best choice of algorithm for this method is still to be resolved, while the best choice for the sparse coding subproblem (1) is much more clear. In this section we provide a brief summary of the algorithms that will be compared. We do not include the methods of [3] and [5] because the methods that are considered here have been shown, in [4] and [9] respectively, to have superior time-convergence performance.

2.1. Iterated Sherman-Morrison

Problem (2) can be written in standard ADMM form as

$$\mathop{\arg\min}_{\{\mathbf{d}_m\},\{\mathbf{g}_m\}} \frac{1}{2}\sum_k \Big\|\sum_m \mathbf{x}_{k,m} * \mathbf{d}_m - \mathbf{s}_k\Big\|_2^2 + \sum_m \iota_{C_{\mathrm{PN}}}(\mathbf{g}_m) \quad \text{s.t.} \quad \mathbf{d}_m = \mathbf{g}_m \;\; \forall m \;. \qquad (3)$$

where $\{\mathbf{g}_m\}$ is an auxiliary variable constrained to be equal to the primary solution variable $\{\mathbf{d}_m\}$, and $\iota_{C_{\mathrm{PN}}}$ is the indicator function of constraint set C_{PN}. The corresponding ADMM updates are

$$\begin{aligned}
\{\mathbf{d}_m\}^{(j+1)} &= \arg\min_{\{\mathbf{d}_m\}} \frac{1}{2}\sum_k \Big\| \sum_m \mathbf{x}_{k,m} * \mathbf{d}_m - \mathbf{s}_k \Big\|_2^2 \\
&\quad + \frac{\sigma}{2}\Big\| \mathbf{d}_m - \mathbf{g}_m^{(j)} + \mathbf{h}_m^{(j)} \Big\|_2^2 , \qquad (4)\\
\{\mathbf{g}_m\}^{(j+1)} &= \arg\min_{\{\mathbf{g}_m\}} \sum_m \iota_{C_{\mathrm{PN}}}(\mathbf{g}_m) \\
&\quad + \frac{\sigma}{2}\Big\| \mathbf{d}_m^{(j+1)} - \mathbf{g}_m + \mathbf{h}_m^{(j)} \Big\|_2^2 , \qquad (5)\\
\mathbf{h}_m^{(j+1)} &= \mathbf{h}_m^{(j)} + \mathbf{d}_m^{(j+1)} - \mathbf{g}_m^{(j+1)} , \qquad (6)
\end{aligned}$$

where $\sigma > 0$ is the penalty parameter. The only computationally expensive step is update (4), which can be expressed as

$$\arg\min_{\hat{\mathbf{d}}} \quad \frac{1}{2}\sum_k \Big\| \hat{X}_k \, \hat{\mathbf{d}} - \hat{\mathbf{s}}_k \Big\|_2^2 + \frac{\sigma}{2}\Big\| \hat{\mathbf{d}} - \hat{\mathbf{g}} + \hat{\mathbf{h}} \Big\|_2^2 , \qquad (7)$$

where $\hat{\mathbf{z}}$ denotes the DFT of variable $\mathbf{z}$, $\hat{X}_{k,m} = \mathrm{diag}(\hat{\mathbf{x}}_{k,m})$, $\hat{X}_k = (\hat{X}_{k,0} \;\; \hat{X}_{k,1} \;\; \ldots)$, and

$$\hat{\mathbf{d}} = \begin{pmatrix} \hat{\mathbf{d}}_0 \\ \hat{\mathbf{d}}_1 \\ \vdots \end{pmatrix} \quad \hat{\mathbf{g}} = \begin{pmatrix} \hat{\mathbf{g}}_0 \\ \hat{\mathbf{g}}_1 \\ \vdots \end{pmatrix} \quad \hat{\mathbf{h}} = \begin{pmatrix} \hat{\mathbf{h}}_0 \\ \hat{\mathbf{h}}_1 \\ \vdots \end{pmatrix} . \qquad (8)$$

The solution is given by the linear system

$$\left(\sum_k \hat{X}_k^H \hat{X}_k + \sigma I \right) \hat{\mathbf{d}} = \sum_k \hat{X}_k^H \hat{\mathbf{s}}_k + \sigma \left(\hat{\mathbf{g}} - \hat{\mathbf{h}} \right) . \qquad (9)$$

Since this system consists of K rank-one terms and a diagonal term, it is much more expensive to solve than the corresponding system that arises in the ADMM solution of the sparse coding problem (1), but an effective solution is still possible by iterated application of the Sherman-Morrison formula [4], which we will refer to as Iterated Sherman-Morrison (ISM). Of course, when $K = 1$ (i.e. a single training image), the system consists of a rank-one term and a diagonal term, and can be efficiently solved by direct application of the Sherman-Morrison formula, as in the sparse coding problem [2].

2.2. Consensus Optimization

The dictionary learning problem is computationally expensive because of the coupling across all K training signals, $\mathbf{s}_k$, introduced by the need to learn a single set of dictionary filters for all images. It can be posed as a global variable consensus problem [8, Ch. 7] by introducing a distinct set of dictionaries for each training signal together with a constraint that these dictionaries are all the same [6]

$$\begin{aligned}
&\arg\min_{\{\mathbf{d}_{k,m}\},\{\mathbf{g}_m\}} \frac{1}{2}\sum_k \left\| \sum_m \mathbf{x}_{k,m} * \mathbf{d}_{k,m} - \mathbf{s}_k \right\|_2^2 + \sum_m \iota_{C_{\mathrm{PN}}}(\mathbf{g}_m) \\
&\qquad \text{s.t} \quad \mathbf{d}_{0,m} = \cdots = \mathbf{d}_{K-1,m} = \mathbf{g}_m \; \forall \; m = 0 , \qquad (10)
\end{aligned}$$

where $\mathbf{d}_{k,m}$ represents the m^{th} filter for the k^{th} image and $\mathbf{g}_m$ the global consensus variable for the m^{th} filter. The ADMM updates are

$$\begin{aligned}
\{\mathbf{d}_{k,m}\}^{(j+1)} &= \arg\min_{\{\mathbf{d}_{k,m}\}} \quad \Big\| \sum_m \mathbf{x}_{k,m} * \mathbf{d}_{k,m} - \mathbf{s}_k \Big\|_2^2 \\
&\quad + \frac{\sigma}{2}\sum_m \Big\| \mathbf{d}_{k,m} - \mathbf{g}_m^{(j)} + \mathbf{h}_{k,m}^{(j)} \Big\|_2^2 , \qquad (11)\\
\{\mathbf{g}_m\}^{(j+1)} &= \arg\min_{\{\mathbf{g}_m\}} \quad \sum_m \iota_{C_{\mathrm{PN}}}(\mathbf{g}_m) \\
&\quad + \frac{\sigma}{2}\sum_m \Big\| \mathbf{d}_{k,m}^{(j+1)} - \mathbf{g}_m + \mathbf{h}_{k,m}^{(j)} \Big\|_2^2 , \qquad (12)\\
\mathbf{h}_{k,m}^{(j+1)} &= \mathbf{h}_{k,m}^{(j)} + \mathbf{d}_{k,m}^{(j+1)} - \mathbf{g}_m^{(j+1)} . \qquad (13)
\end{aligned}$$

Again, by defining $\hat{X}_{k,m}$ and $\hat{X}_k$ as before, and defining

$$\hat{\mathbf{d}}_k = \begin{pmatrix} \hat{\mathbf{d}}_{k,0} \\ \hat{\mathbf{d}}_{k,1} \\ \vdots \end{pmatrix} \quad \hat{\mathbf{g}} = \begin{pmatrix} \hat{\mathbf{g}}_0 \\ \hat{\mathbf{g}}_1 \\ \vdots \end{pmatrix} \quad \hat{\mathbf{h}}_k = \begin{pmatrix} \hat{\mathbf{h}}_{k,0} \\ \hat{\mathbf{h}}_{k,1} \\ \vdots \end{pmatrix} , \qquad (14)$$

update (11) reduces to solving the linear system

$$\left(\hat{X}_k^H \hat{X}_k + \sigma I \right) \hat{\mathbf{d}}_k = \hat{X}_k^H \hat{\mathbf{s}}_k + \sigma \left(\hat{\mathbf{g}} - \hat{\mathbf{h}}_k \right) , \qquad (15)$$

which is independent for each signal, and can be solved by a single application of the Sherman-Morrison formula [2, 4]. Update (12) is solved by [8, Ch. 7]

$$\mathbf{g}_m^{(j+1)} = \mathrm{prox}_{\iota_{C_{\mathrm{PN}}}} \left(\frac{1}{K} \sum_k \left(\mathbf{d}_{k,m}^{(j+1)} + \mathbf{h}_{k,m}^{(j)} \right) \right) . \qquad (16)$$

2.3. 3D

If the set of signals is consolidated along an additional dimension and the convolutions are computed taking this extra dimension into account, all computations can proceed as if just one signal is being represented, i.e. $K = 1$, so that (9) can be solved via the Sherman-Morrison formula [6]. For example, if the signals correspond to images, the entire training set is considered to form a single 3D image, with corresponding 3D coefficient maps and 3D dictionary filters, with convolutions computed in all three dimensions. The implicit zero-padding of the filters $\{\mathbf{d}_m\}$ to the size of the coefficient maps $\{\mathbf{x}_m\}$ includes the extra dimension to ensure that the filters have an appropriately constrained support in the spatial domain of the original signals.

3. SUBPROBLEM COUPLING

In constructing a dictionary learning algorithm from the sparse coding (1) and dictionary subproblems (2), Bristow et al. [3] and Heide et al. [5] define a single Augmented Lagrangian, directly yielding a set of six update equations for both the coefficient maps and dictionary filters, and automatically determining the primary variables as the coupling variables, i.e. use of coefficient maps $\{\mathbf{x}_{k,m}\}$ in the dictionary subproblem and dictionary filters $\{\mathbf{d}_m\}$ in the sparse coding subproblem.

In contrast, when Augmented Lagrangians are defined independently for the sparse coding and dictionary learning subproblems [4, 6], the choice of coupling variables does not follow directly from the derivation. The most natural choice might seem to be to replicate that imposed by the single Augmented Lagrangian derivation, but some experiments involving the ISM dictionary update found that this coupling was less stable, being more sensitive to the choice of penalty parameters ρ and σ for the sparse coding and dictionary subproblem ADMM algorithms respectively, and resulted in slower convergence than a coupling using only auxiliary variables [4].

Šorel et al. do not address this issue at all in their published work [6], but from inspection of their publicly available software [10] it appears that they use the primary coefficient map variable and the auxiliary dictionary variable; presumably, like most other authors, they wished to follow the example of [3] in using only primary variables, but were forced to use the auxiliary dictionary variable since the primary variable in the 3D and consensus updates is not suitable for directly passing to the sparse coding subproblem.

In the following, we will refer to a choice of coupling variables for a dictionary learning algorithm by a pair constructed from the choice made for the coefficient maps and for the dictionary variables, in that order. For example, a primary-auxiliary coupling indicates that the primary coefficient map variable from the sparse coding subproblem is passed to the dictionary subproblem, and the auxiliary dictionary variable from the dictionary subproblem is passed to the sparse coding subproblem. These labels will be abbreviated using "P" to indicate primary and "A" to indicate auxiliary, e.g. primary-auxiliary is abbreviated as PA.

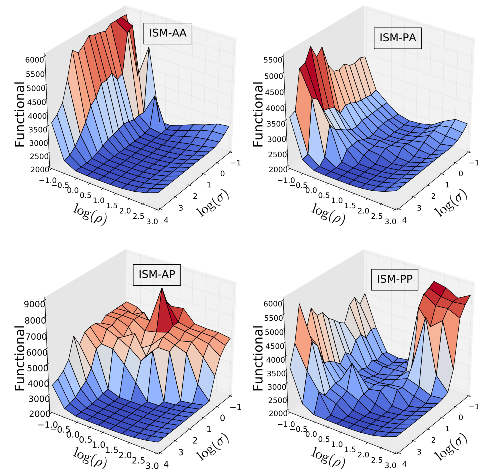

Fig. 1. Grid search surfaces for different couplings of the ISM dictionary subproblem algorithm with $K = 20$ and after 100 iterations. Each surface represents the value of the CBPDN functional (1) obtained for different parameters ρ and σ.

4. RESULTS

We consider two distinct sets of experiments. In the first set we compare the performance of dictionary learning algorithms constructed using different couplings and different dictionary subproblem algorithms on a training set of 20 images. The ISM dictionary subproblem algorithm is used with all four possible couplings (AA, AP, PA, and AP), while the 3D and consensus dictionary subproblem methods are used with the AA and PA couplings since the primary variables of these algorithms are not directly compatible with the interleaved update framework that we consider (e.g., the primary consensus dictionary subproblem variable consists of a different dictionary per training image). In the second set of experiments we select the best coupling for each of the three dictionary subproblem algorithms and compare their performance over training sets of different sizes.

Our training image set consists of greyscale images of size 256 $\times$ 256 pixels, cropped and rescaled from a set of images with a Creative Commons license obtained from Flickr. The 8 bit greyscale images are divided by 255 so that pixel values are within the interval [0,1], and they are pre-processed with a highpass filtering step which is a common approach for convolutional sparse representations [11, 12, 4]. We learn a dictionary of 64 filters of size 8 $\times$ 8 for sets of 5, 10, 20 and 40 images, setting the sparsity parameter $\lambda = 0.1$. All the results reported here were computed using the Python implementation of the SPORCO library [13] on a Linux workstation equipped with two Xeon E5-2690V4 CPUs.

Since convergence rates depend critically on the selection of suitable penalty parameters ρ (sparse coding) and σ (dictionary subproblem), we implemented a grid search over the space of penalty parameters. First, to establish a common starting point, we ran 50 iterations of the ISM method for the set of 20 images and used the resulting partially trained dictionary as the starting point for the parameter search for all the methods[2]. We then did a parameter search over logarithmically spaced grids with the following resolutions: ρ, 10 logarithmically spaced points in $[0.1, 1000]$; and σ, 15 logarithmically spaced points in $[0.1, 10000]$. The best set of (ρ, σ) for each combination of method and interleaved update, i.e. the ones yielding the lowest value of the CBPDN functional at 100 iterations, were selected as the optimal parameters. For illustration, the grid search result for the 20 images set using ISM is shown in Fig. 1.

This procedure was repeated for sets of 1, 5, 10 and 20 images. Since the cost of the grid search grows rapidly with the number of images, we decided to set the parameters for the 40 images case by extrapolating from the best values computed for the other cases. A summary of the optimal parameters computed, or extrapolated, for the AA interleaved update is displayed in Fig. 2 (Left).

All the methods have different optimal parameters, but in all cases the surface generated is flatter for a wider range of the penalty parameter, σ, for the dictionary subproblem than the range of the penalty parameter, ρ, for the sparse coding subproblem, leading us to conclude that the methods are more sensitive to the setting of ρ. This observation seems to agree with the qualitative behavior reported on [14, Sec. 4], the only other work to make a systematic exploration of the parameter space, albeit only for a single update method and coupling choice. This work considers the Bristow et al. formulation [3], concluding that there are parameter regions with erratic behavior (very small values of ρ and σ), some where the objective function increases rapidly (large ρ and σ values) and some with a smooth behavior (medium ρ and σ values). The ranges reported are closer to our ISM-PP, since the Bristow et al. formulation is a primary-primary coupling. The actual ranges for the other types of couplings are different, but the qualitative behavior is the same (see Fig. 1). Note that the scaling of σ for ISM depends on K.

We compare the performance of the differently coupled schemes using the set of 20 images. Results for all the combinations of methods and update couplings discussed here are shown in Fig. 3. It is clear that the worst performance is obtained for the PP coupling and the best for the AA couplings. The performance for the AP coupling is close to the AA, while the performance of the PA coupling is worse. It appears that using soft-thresholded coefficient maps $\{\mathbf{y}_{k,m}\}$ in the dictionary subproblem is more critical to obtaining good performance than using projected dictionary filters $\{\mathbf{g}_m\}$ in

[2] The parameters used for this initialization were also tuned by searching on a logarithmic grid with resolution 10 points in [0.1, 1000] per parameter, and selecting the ones yielding the lowest CBPDN functional at 50 iterations: $\rho = 2.15$ and $\sigma = 16.68$.

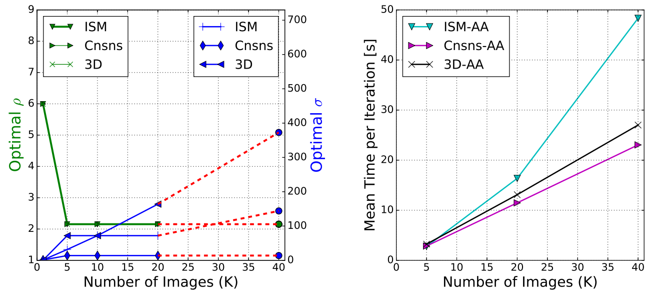

Fig. 2. (**Left**) Comparison of grid search parameters for auxiliary variable alternation (AA) for sets of 1, 5, 10 and 20 images. Parameters are extrapolated for set of 40 images. (**Right**) Comparison of time per iteration for the three dictionary learning methods with auxiliary variable alternation (AA) for sets of 5, 20 and 40 images.

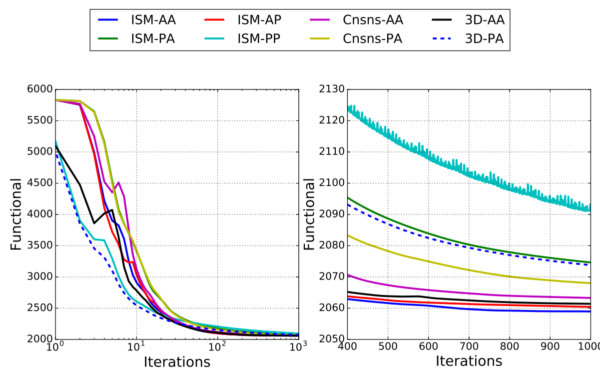

Fig. 3. A comparison on a set of 20 images of the decay of the value of the CBPDN functional (1) with respect to iterations, for four interleaved combinations for iterated Sherman Morrison (ISM), and primary and auxiliary couplings in sparse coding for global consensus (Cnsns) and 3D (3D).

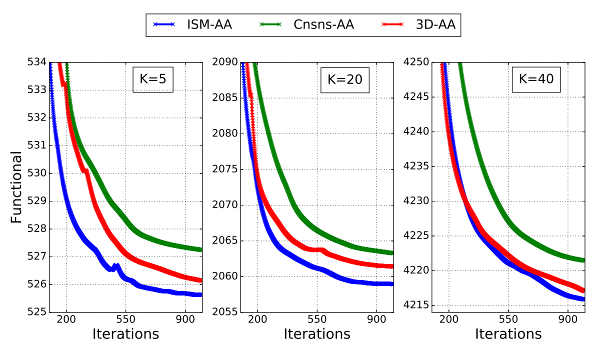

Fig. 4. A comparison on sets of 5, 20 and 40 images of the decay of the value of the CBPDN functional (1) with respect to iterations, for auxiliary variable alternation (AA).

the sparse coding subproblem. This suggests that obtaining the correct support for non-zero values in the coefficient maps is a critical factor. With respect to the consensus and 3D dictionary subproblem algorithms, note that the AA coupling gives significantly better behaviour than the PA coupling used in [6].

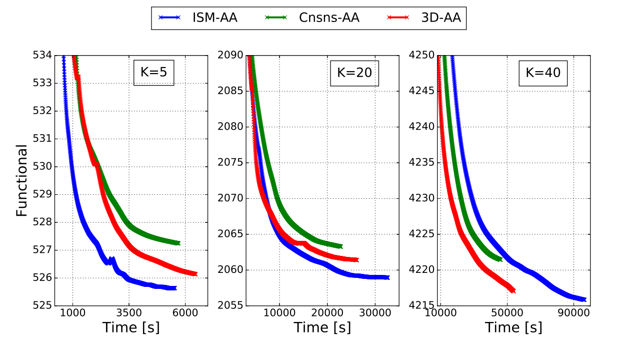

Fig. 5. A comparison on sets of 5, 20 and 40 images of the decay of the value of the CBPDN functional (1) with respect to run time, for auxiliary variable alternation (AA).

In methods that use primary forms of coupling, it is customary to perform a number of iterations in each subproblem before exchanging variables between them [5, 6]. This seems to improve the stability and convergence properties. However, our results indicate that the most direct way to improve the convolutional dictionary learning performance is to use the auxiliary sparse coding variable for coupling the subproblems (i.e. AA or AP variants).

In order to evaluate the performance of the algorithms with respect to the number of images in the training set, we learned dictionaries using sets of 5, 20 and 40 images. We used AA couplings since they yield better results, and tuned the penalty parameters for each case following the parameter search procedure described before. The results obtained after running each method for 1000 iterations are included in Figs. 4 and 5. ISM consistently achieves the lowest functional value, but it has a poor time scaling for a large number of images. For large K both consensus and 3D achieve a good functional value substantially faster than ISM. Also, note that 3D has consistently better performance than consensus, at the cost of a larger memory requirement.

Finally, Fig. 2 (Right) displays the scaling of the mean time per iteration for the methods using auxiliary couplings and sets of 5, 20 and 40 images. Note that consensus and 3D methods exhibit a linear scaling with the number of images, while the scaling of the ISM method becomes worse for an increasing set of images. This behavior reflects the expected time complexity of the methods: $\mathcal{O}(MKN\log N + K^2MN)$ for ISM, $\mathcal{O}(MKN\log N + MKN)$ for consensus, and $\mathcal{O}(MKN(\log N + \log K) + MKN)$ for 3D.

5. CONCLUSION

Our careful comparison of coupling strategies for convolutional dictionary learning indicates that the auxiliary-auxiliary coupling, which is the least widely used, provides the best performance independent of the dictionary subproblem algorithm. It is also interesting to note that the performance is more sensitive to the choice of coefficient map variable than it is to the choice of the dictionary variable. With respect to the dictionary subproblem algorithm, ISM [4] consistently provides the lowest functional value by iteration count, and is also the fastest for relatively small K, but for larger K values both consensus [6] and 3D [6] are substantially faster, with 3D giving the best overall performance. It is worth noting that the dictionary learning algorithms constructed from the consensus and 3D algorithms using the AA coupling, as proposed here, appear to give significantly better performance than the PA couplings used by Šorel et al. [6].

6. REFERENCES

[1] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks," in *Proc. IEEE Conf. Comp. Vis. Pat. Recog. (CVPR)*, Jun. 2010, pp. 2528–2535. doi:10.1109/cvpr.2010.5539957

[2] B. Wohlberg, "Efficient convolutional sparse coding," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2014, pp. 7173–7177. doi:10.1109/ICASSP.2014.6854992

[3] H. Bristow, A. Eriksson, and S. Lucey, "Fast convolutional sparse coding," in *Proc. IEEE Conf. Comp. Vis. Pat. Recog. (CVPR)*, Jun. 2013, pp. 391–398. doi:10.1109/CVPR.2013.57

[4] B. Wohlberg, "Efficient algorithms for convolutional sparse representations," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 301–315, Jan. 2016. doi:10.1109/TIP.2015.2495260

[5] F. Heide, W. Heidrich, and G. Wetzstein, "Fast and flexible convolutional sparse coding," in *Proc. IEEE Conf. Comp. Vis. Pat. Recog. (CVPR)*, Jun. 2015, pp. 5135–5143. doi:10.1109/CVPR.2015.7299149

[6] M. Šorel and F. Šroubek, "Fast convolutional sparse coding using matrix inversion lemma," *Digital Signal Processing*, vol. 55, pp. 44–51, 2016. doi:10.1016/j.dsp.2016.04.012

[7] K. Engan, S. O. Aase, and J. H. Husøy, "Method of optimal directions for frame design," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, vol. 5, 1999, pp. 2443–2446. doi:10.1109/icassp.1999.760624

[8] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010. doi:10.1561/2200000016

[9] B. Wohlberg, "Boundary handling for convolutional sparse representations," in *Proc. IEEE Conf. Image Process. (ICIP)*, Phoenix, AZ, USA, Sep. 2016, pp. 1833–1837. doi:10.1109/ICIP.2016.7532675

[10] M. Šorel and F. Šroubek, Software available from https://github.com/michalsorel/convsparsecoding, 2016.

[11] K. Kavukcuoglu, P. Sermanet, Y. Boureau, K. Gregor, M. Mathieu, and Y. LeCun, "Learning convolutional feature hierarchies for visual recognition," in *Advances in Neural Information Processing Systems (NIPS 2010)*, 2010, pp. 1090–1098.

[12] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *Proc. IEEE Int. Conf. Comp. Vis. (ICCV)*, Barcelona, Spain, Nov. 2011, pp. 2018–2025. doi:10.1109/iccv.2011.6126474

[13] B. Wohlberg, "SParse Optimization Research COde (SPORCO)," Software library available from http://purl.org/brendt/software/sporco, 2016.

[14] B. Kong and C. C. Fowlkes, "Fast convolutional sparse coding (FCSC)," University of California, Irvine, Tech. Rep., May 2014.