BLOCK ORTHONORMAL OVERCOMPLETE DICTIONARY LEARNING

Cristian Rusu ⋆† *and Bogdan Dumitrescu* †

⋆ IMT Institute for Advanced Studies Lucca
Piazza San Ponziano 6, 55100, Lucca, Italy (e-mail: cristian.rusu@imtlucca.it)
† University "Politehnica" of Bucharest
Department of Automatic Control and Computers
Spl. Independenţei 313, Bucharest 060042, Romania (e-mail: bogdan.dumitrescu@acse.pub.ro)

ABSTRACT

In the field of sparse representations, the overcomplete dictionary learning problem is of crucial importance and has a growing application pool where it is used. In this paper we present an iterative dictionary learning algorithm based on the singular value decomposition that efficiently construct unions of orthonormal bases. The important innovation described in this paper, that affects positively the running time of the learning procedures, is the way in which the sparse representations are computed - data are reconstructed in a single orthonormal base, avoiding slow sparse approximation algorithms - how the bases in the union are used and updated individually and how the union itself is expanded by looking at the worst reconstructed data items. The numerical experiments show conclusively the speedup induced by our method when compared to previous works, for the same target representation error.

***Index Terms*—** sparse representations, orthogonal blocks, overcomplete dictionary learning.

1. INTRODUCTION

Tools from the sparse representations [1] [2] field have been extensively used to model various observed data, e.g. audio and image, in terms of sparse linear models with numerous practical applications [3].

In this field, one of the central open problems is developing overcomplete dictionary learning algorithms. This approach suggests a different way of considering linear transformations: instead of using a popular, well-known transform (such as Fourier or wavelet) to represent the data, a custom transform can be learned by analyzing a small, but relevant, available dataset. The goal is that, given the dataset $\boldsymbol{Y} \in \mathbb{R}^{n\times N}$ we construct the factorization $\boldsymbol{Y} \approx \boldsymbol{D}\boldsymbol{X}$ where the matrix $\boldsymbol{D} \in \mathbb{R}^{n\times m}$ is called dictionary, the normalized columns $\boldsymbol{d}_j \in \mathbb{R}^n$ with $j = 1,\dots,m$, are called atoms and the sparse representations matrix $\boldsymbol{X} \in \mathbb{R}^{m\times N}$. Notice that the problem dimensions follow $n \le m \ll N$.

Stated as an optimization problem, given the dataset $\boldsymbol{Y}$ and the target sparsity s_0 we can define the overcomplete dictionary learning problem as

$$\begin{aligned} \underset{\boldsymbol{D},\boldsymbol{X}}{\text{minimize}} \quad & \|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_F^2 \\ \text{subject to} \quad & \|\boldsymbol{x}_i\|_0 \le s_0,\ 1\le i\le N \\ & \|\boldsymbol{d}_j\|_2 = 1,\ 1\le j\le m, \end{aligned} \tag{1}$$

where $\|\boldsymbol{x}\|_0$ is the ℓ_0 pseudo-norm (the number of non-zero components in the vector $\boldsymbol{x}$). This problem is hard due to the bilinear objective function and the N constraints that are NP-hard.

Numerous methods that tackle problem (1) have been proposed (such as K-SVD [4], AK-SVD [5], LS-DLA [6]) and the general approach is to solve it by alternative iterations: keep the dictionary fixed and find the sparse representations by applying an algorithm such as Orthogonal Matching Pursuit (OMP) [7] and then, with the new representations, update the dictionary (the whole at once or one atom at a time). Current state of the art methods work well in practice, especially in terms of performance (low representation errors).

The general dictionary learning problem does not assume any structure for the dictionary $\boldsymbol{D}$. In this paper we assume that the dictionary is a union of orthonormal bases (ONBs) [8] - or, in the particular case, that the dictionary is orthonormal. Consider now the problem

$$\begin{aligned} \underset{\boldsymbol{D},\boldsymbol{X}}{\text{minimize}} \quad & \|\boldsymbol{Y}-[\boldsymbol{Q}_1\ \dots\ \boldsymbol{Q}_L]\boldsymbol{X}\|_F^2 \\ \text{subject to} \quad & \|\boldsymbol{x}_i\|_0 \le s_0,\ 1\le i\le N \\ & \boldsymbol{Q}_j^T\boldsymbol{Q}_j = \boldsymbol{I}_n, 1\le j\le L, \end{aligned} \tag{2}$$

where the dictionary $\boldsymbol{D}$ is written explicitly as a union of L ONBs noted $\boldsymbol{Q}_j \in \mathbb{R}^{n\times n}$ with $j = 1,\dots,L$.

Prior dictionary learning methods that work with unions of ONBs follow the same iterative idea as general dictionary

This work was supported by the Romanian National Authority for Scientific Research, CNCS - UEFISCDI, project number PN-II-ID-PCE-2011-3-0400.

learning methods do but operate with the Singular Value Decomposition (SVD) in order to build naturally the orthonormal structures. In order to construct the sparse representations, the OMP algorithm is used again. Each individual orthonormal block is updated after analyzing the single block case, which plays a crucial role in this learning framework. Setting the solver structure aside, constraining the dictionary to be a union of ONBs has a true practical significance - for example, images can be modeled as a superposition of several layers that have sparse representations in particular ONBs [9].

Contribution. In this paper we propose a new approach to the union of ONBs dictionary learning problem. We propose a new way of constructing the unions of ONBs by changing the way the sparse representations are created. Unlike the previous methods, that use atoms from any orthonormal block, we attach each data item to the orthobase that best represents it avoiding thus the utilization of the OMP algorithm and speeding up the direct and inverse dictionary applications. From this point of view, the method proposed in this paper is similar, in nature, to the block sparse dictionary learning methods [10] albeit with some important differences: each block is an orthobase and only one block is used in each representation. This novel structure allows for the fast application of the dictionary. When training the blocks of the dictionary, each block is constructed using a well-known orthonormal dictionary learning method applied only on the training data associated with this particular block, while the dictionary expands in a greedy fashion by adding at each step an extra block.

The method proposed in this paper is slightly different than the common wisdom presented in the field of sparse representations. There, the general principle is to allow, for the same target signal, the representation atoms to originate from different bases. This comes with great representation success but at the expense of relatively large running times which are mainly the fault of the slow sparse approximation algorithms. If until now, the diversity in the dictionary was generated by atom directions, in our case the diversity is generated by allowing for a large number of independent orthobases (that can be operated very fast).

The manuscript is organized as follows: Section 2 outlines the current approach to the ONBs dictionary learning problem, Section 3 describes the new approach that we consider while Section 4 outlines some numerical experiments that validate the new method. Section 5 concludes the paper and sets future objectives in this research direction.

2. PRIOR ART

The results presented in this section are based on [8], a paper that describes an efficient dictionary learning method based on unions of ONBs. The new results presented in this paper build upon this work.

In the first case we consider that the dictionary is composed of a single orthonormal base. The problem is easily stated as

$$\begin{aligned} \underset{\boldsymbol{Q},\boldsymbol{X}}{\text{minimize}} \quad & \|\boldsymbol{Y}-\boldsymbol{Q}\boldsymbol{X}\|_F^2 \\ \text{subject to} \quad & \|\boldsymbol{x}_i\|_0 \le s_0,\ 1 \le i \le N \\ & \boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I}_n. \end{aligned} \tag{3}$$

The crucial observation is that, in the iterative process to solve this problem when the sparse representations matrix $\boldsymbol{X}$ is kept fixed a singular value decomposition is used for the update. Concretely, each iteration of the orthonormal dictionary learning algorithm is presented next.

1ONB Alternative optimization iteration. Given the dataset $\boldsymbol{Y} \in \mathbb{R}^{n\times N}$, the target sparsity s_0 and the maximum number of iterations K_0, train the orthonormal dictionary $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$ such that $\|\boldsymbol{Y}-\boldsymbol{Q}\boldsymbol{X}\|_F$ is reduced by iterating the alternative steps:

1. With the fixed dictionary $\boldsymbol{Q}$, compute the sparse representations with target sparsity s_0 by computing $\boldsymbol{X} = \boldsymbol{Q}^T\boldsymbol{Y}$ and keeping the largest s_0 absolute value entries on each column.

2. Using the new matrix $\boldsymbol{X}$, the new orthonormal dictionary is computed via the SVD decomposition

$$\boldsymbol{Y}\boldsymbol{X}^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{4}$$

and the updated dictionary is taken to be

$$\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{V}^T. \tag{5}$$

Details regarding this approach are presented in [8]. Notice that each data item can use atoms originating from different orthobases. This definitely allows for better sparse approximation at the expense of using the slow OMP algorithm that cannot exploit the orthogonal structure of the dictionary. With a good initialization, the procedure needs to run only for a few iterations to reach very good results ($K_0 < 5$).

Based on the previous result, we present the general training algorithm for the union of L ONBs.

Union of L orthonormal bases dictionary learning (UONB). Given the dataset $\boldsymbol{Y} \in \mathbb{R}^{n\times N}$, the target sparsity s_0, the number of orthobases L and the maximum number of iterations K, construct the dictionary $\boldsymbol{D} = [\boldsymbol{Q}_1 \ \dots \ \boldsymbol{Q}_L] \in \mathbb{R}^{n\times Ln}$, column-wise concatenated, and the sparse representations matrix $\boldsymbol{X} \in \mathbb{R}^{Ln\times N}$ such that $\|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_F$ is reduced.

- **Initialization:**

1. Choose an initial dictionary $\boldsymbol{D}_0$ (a very good starting point can be $\boldsymbol{U}$ from $\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, a SVD decomposition on the whole dataset).

2. Update the sparse representation using the OMP algorithm to construct $\boldsymbol{X} = [\boldsymbol{X}_1^T \ \dots \ \boldsymbol{X}_L^T]^T$, row-wise concatenated.

• **Iterations:** for $k = 1, \dots, K$

1. Update the dictionary. For each orthobase $i = 1, \dots, L$
   (a) Extract the working dataset: $\boldsymbol{Y}_i = \boldsymbol{Y} - \sum_{j \neq i} \boldsymbol{Q}_j \boldsymbol{X}_j$.
   (b) Compute the SVD: $\boldsymbol{Y}_i \boldsymbol{X}_i = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$.
   (c) Update the dictionary: $\boldsymbol{Q}_i = \boldsymbol{U}\boldsymbol{V}^T$.
2. Compute $\boldsymbol{X}$, the new representations matrix, using again the OMP algorithm.
3. Possibly check additional stopping criterion.

This method works well in practical applications but it is relatively slow. The focus of the paper herein is to improve the running time of the sparse reconstruction strategy.

3. THE PROPOSED METHOD

In this section we present a different approach to the union of ONBs dictionary learning problem than the one described earlier. We propose a new way to construct the sparse representations such that the OMP algorithm is avoided, and thus the running time of the learning procedure is greatly reduced.

One of the biggest issues with the current state-of-the-art dictionary learning algorithms is the poor running time. This is what we address in this section of the paper. We identify the source of the problem as being the usage of slow sparse reconstruction algorithms, like the Orthogonal Matching Pursuit (OMP). The utilization of algorithms that are faster than OMP is not actually a viable option since these methods have, in the general case, a series of undesirable properties (e.g. they perform much worse in terms of representation error, situations can be constructed where the same atoms are selected multiple times in the same iterative process). The only situation where sparse reconstruction algorithms can be avoided is the orthonormal case. When dealing with an orthonormal dictionary, the sparse approximation problem reduces to computing the projections of the target signal and keeping the largest absolute value entries. To select the largest entries, per signal, a fast partial sorting algorithm [11] is used whose complexity is only $\mathrm{O}\big(n\big)$.

Taking this observation into account we propose a new strategy for the construction of the sparse representations in a union of ONBs: instead of using atoms from different ONBs, we attach each data item to a particular ONB and use only its atoms for the reconstruction. This way, we group data items together that can have a sparse construction in a single ONB. We construct an algorithm that at each step grows the current dictionary with one more ONB that is trained by looking at the worst constructed data items from the available dataset. Overall, these observations play an important role when looking at the scaling capabilities of the proposed method.

The algorithm, called Single Block Orthogonal Dictionary Learning (SBO), is presented next in detail.

Single Block Orthogonal Dictionary Learning (SBO). Given the dataset $\boldsymbol{Y} \in \mathbb{R}^{n \times N}$, the target sparsity s_0, the target representation error ϵ, construct the dictionary $\boldsymbol{D} = [\boldsymbol{Q}_1 \ \dots \ \boldsymbol{Q}_M] \in \mathbb{R}^{n \times Mn}$ and the sparse representations matrix $\boldsymbol{X} = [\boldsymbol{X}_1^T \ \dots \ \boldsymbol{X}_M^T] \in \mathbb{R}^{Mn \times N}$ that allocates each data item to a single ONB such that $\|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_F$ is reduced.

• **Initialization:**

1. Iteratively train M_0 individual ONBs by randomly selecting each time $P_0\%$ of the available dataset and applying the 1ONB iterations K_0 times.
2. Allocate each data item $\boldsymbol{y}_i, i = 1, \dots, N$ to only one of the previously computed ONB $\boldsymbol{Q}_{m_i^\star}$ by:

$$m_i^\star = \underset{m=1,\dots,M_0}{\text{argmax}} \sum_{k=1}^{s_0} |\boldsymbol{Q}_m^T \boldsymbol{y}_i|_{[k]}, \tag{6}$$

   where $\boldsymbol{z}_{[k]}$ stands for the k^{th} largest component of $\boldsymbol{z}$. Indices allocated to $\boldsymbol{Q}_m$ are denoted by the set $\mathcal{I}_m$.

• **Iterations:**

1. Construct the set of indices:

$$\mathbb{W} = \left\{ \frac{1}{\|\boldsymbol{y}_i\|_2^2} \sum_{k=1}^{s_0} |\boldsymbol{Q}_{m_i^\star}^T \boldsymbol{y}_i|_{[k]}^2 \right\}_{<1,\dots,\lceil PN \rceil>}, \forall i \tag{7}$$

   where $\boldsymbol{z}_{<k>}$ stands for the index of the k^{th} smallest component of $\boldsymbol{z}$, $m_i^\star$ is the current allocation of $\boldsymbol{y}_i$ and train using 1ONB initialized with the SVD result $\boldsymbol{U}$, $\boldsymbol{Y}_{\mathbb{W}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$. Add this new orthobase to the current union ($M = M + 1$).
2. Use (6) to produce the new sets $\mathcal{I}_m, m = 1, \dots, M$.
3. Since the union can be treated as a queue, in a last in first out (LIFO) order, update each ONB using 1ONB:

$$[\boldsymbol{Q}_m, \boldsymbol{X}_m^T] = \text{1ONB}(\boldsymbol{Y}_{\mathcal{I}_m}, K_0), m = M, \dots, 1. \tag{8}$$

4. Check stop condition: $\|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\| \leq \epsilon$.

A discussion on each step of SBO follows.

In the initialization step we construct M_0 orthonormal dictionaries using a randomly selected fraction ($P_0\%$) of the dataset. Typical values of M_0 are included in the range $[1, \dots, 5]$ but this choice should also consider the total dimension of the dataset and the target sparsity imposed. For large datasets or when the target sparsity is large (relative close

Table 1: Representation errors and training running times (in seconds) achieved by UONB and SBO.

L	s_0	ϵ_{UONB}	t_{UONB} (s)	ϵ_{SBO}	t_{SBO} (s)	M
3	4	0.676	12.9	0.672	3.9	10
	8	0.418	16.3	0.417	6.8	13
	16	0.233	26.6	0.231	12.0	16
4	4	0.640	15.0	0.639	5.7	12
	8	0.394	21.0	0.394	10.6	17
	16	0.224	36.2	0.222	14.1	18
5	4	0.613	17.6	0.611	9.7	16
	8	0.378	25.9	0.376	13.1	20
	16	0.212	48.7	0.210	19.5	22
6	4	0.591	21.8	0.591	11.1	19
	8	0.366	35.2	0.363	16.5	23
	16	0.200	71.8	0.199	20.6	26

to the working dimension n) experimental runs show that a good choice is to select larger values of M_0. In the allocation step, each data item is attributed to one of the orthobases - the one where the item has the smallest reconstruction error. Since we are dealing with orthobases (energy is preserved), the representation residuals need not be computed. It suffices to compute the energy of the representation coefficients from $\boldsymbol{X}$ and select the orthobase where this energy is highest. This initialization step is very important since in general it has an important impact on the total number M of bases that will be added to the dictionary.

Each iteration serves to expand the union of ONBs such that the overall representation error is reduced. Each iteration starts by identifying the $P\%$ worst reconstructed data items from the dataset. This set is the working set for the construction of a new orthobase to expand the union. In some sense, the dictionary is built, gradually, in a greedy manner that should lead to a significant reduction in the representation error. An expansion idea in this spirit also works well for general dictionary learning algorithms [12]. After the new orthobase is added to the union, the allocation process is repeated. Here we expect the worst constructed data items to be attached to the new constructed base. Since the new allocation may have changed the data pool of every orthobase, we check and train for K_0 iterations each such base allowing it to adapt to its newly allocated data items. Each iteration checks if the target total representation error was reached and the algorithm stops accordingly.

Clearly, the dictionaries designed by SBO will compensate the constraint imposed on the reconstruction by producing larger dictionaries. From this point of view, to be competitive with UONB, we expect that $Ls_0 > M$. The inequality comes from comparing the computational complexity of the OMP algorithm $\mathrm{O}\big(Ls_0n^2\big)$ with the complexity of the individual block orthonormal representation $\mathrm{O}\big(Mn^2\big)$.

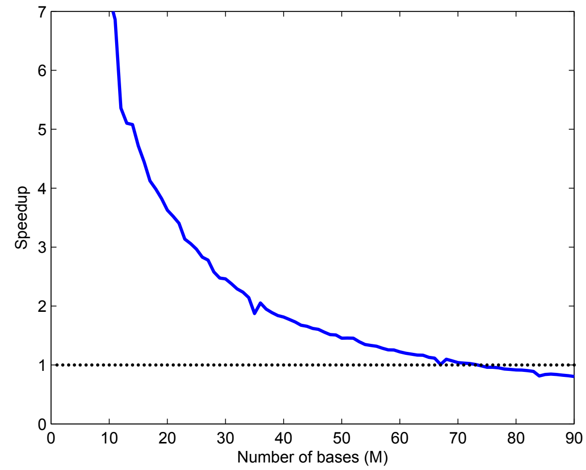

Fig. 1: Speedup ($t_{\text{OMP}}/t_{\text{ALLOC}}$) for the sparse approximation step offered by the block orthonormal algorithm when compared against OMP for various sizes M ($N = 5000$, $L = 3$ and $s_0 = 8$).

4. RESULTS

In this section we describe numerical experiments to validate the method proposed in this paper. Comparisons are provided with the UONB method presented in [8]. When using UONB, in the sparse approximation step where OMP is used we utilize the, publicly available, library OMP-box outlined in [5].

The test data consists of image data extracted from popular test images (such as: lena, peppers, boat). Simulations are executed on a random sample of $N = 5000$ normalized patches of size 8×8 extracted from these images.

In each experimental run, we first apply the UONB algorithm to produce a target representation error that is then provided to the proposed SBO method. In terms of performance we are interested in the following indicators:

1. ϵ_{UONB}, ϵ_{SBO} - the representation errors achieved by the UONB and the SBO algorithms relative to the number of data items N.

2. M - the number of ONBs needed by SBO to reach the representation error of UONB.

3. t_{UONB}, t_{SBO} - the running times of the UONB and the SBO algorithms. We also denote t_{OMP} and t_{ALLOC} the running times of the OMP algorithm and of allocation procedure (6) respectively.

We consider different sparsity levels s_0 and different union lengths L for UONB. In all cases, the number of iterations per single ONB training is $K_0 = 3$, the number of iterations for UONB is $K = 25$ and the number of initial orthobases for SBO is $M_0 = 5$. In the case of SBO, the dimension parameters are taken $P_0 = 10\%$ and $P = 3\%$, chosen after some numerical simulations were conducted. The performance (design dimensions, representation errors and running times) of SBO is depicted in Table 1.

Since we have concluded that, overall, the SBO training procedure is faster than UONB we now focus only on the sparse approximation step. We are interested to see the running time of OMP versus the running time of orthonormal block approximation. In this context, to make a fair comparison, we use a Matlab implementation of OMP, with Cholesky decomposition, instead of the implementation provided by the authors of [5] that is a C compiled version. The results obtained over a selection of $N = 5000$ data items are presented in Figure 1. An important obvious observation is that running time equality is achieved for a very large $M = 72$, when compared to the benchmark $Ls_0 = 24$. This seems to be a consequence of the simplicity of the proposed solution and its ability to take full advantage of simple, well studied, operations (namely block matrix multiplication, sorting).

5. CONCLUSIONS

In this paper we have proposed a new method for constructing unions of orthonormal bases that is significantly faster than the previous approaches. The main idea presented in this paper is the development that allows avoiding sparse approximation algorithms that are relatively slow (like OMP). This is achieved by using in the approximation step only single orthonormal bases. At the cost of a larger dictionary, we develop constructions where each data item of the dataset is allocated to the orthobase where its representation error is lowest. Experimental runs clearly show that the proposed coding mechanism is faster for similar representation capabilities. Future work should address the issue of memory usage when using such large dictionaries or the development of an efficient pruning mechanism to lower their size.

6. ACKNOWLEDGMENT

The authors would like to thank Sotirios A. Tsaftaris for insightful comments on the manuscript.

7. REFERENCES

[1] A. M. Bruckstein, D. L. Donoho and M. Elad, From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images, SIAM Review, vol. 51, no. 1, pp. 34–81, 2009.

[2] D. L. Donoho, Compressed sensing, IEEE Transactions on Information Theory, vol. 52, no. 4, pp. 1289–1306, 2006.

[3] R. G. Baraniuk, E. J. Candes, M. Elad and Ma Yi, Applications of Sparse Representation and Compressive Sensing, Proceedings of the IEEE, vol. 98, no. 6, pp. 906–909, 2010.

[4] M. Aharon, M. Elad and A. Bruckstein, K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation, IEEE Trans. Signal Process., vol. 54, no. 11, pp. 4311–4322, 2006.

[5] R. Rubinstein, M. Zibulevsky and M. Elad, Efficient Implementation of the K-SVD Algorithm using Batch Orthogonal Matching Pursuit, Technical Report - CS Technion, 2008.

[6] K. Skretting, K. Engan and J. H. Husoy, Family of iterative LS-based dictionary learning algorithms, ILS-DLA, for sparse signal representation, Digital Signal Processing, vol. 17, pp. 32–49, 2007.

[7] S. G. Mallat and Z. Zhang, Matching Pursuits With Time-Frequency Dictionaries, IEEE Trans. Signal Process., vol. 41, no. 12, pp. 3397–3415, 1993.

[8] S. Lesage, R. Gribonval, F. Bimbot and L. Benaroya, Learning unions of orthonormal bases with thresholded singular value decomposition, ICASSP, vol. 5, pp. 293–296, 2005.

[9] J. L. Starck, M. Elad and D. L. Donoho, Image decomposition via the combination of sparse representations and a variational approach, IEEE Trans. Imag. Process., vol. 14, no. 10, pp. 1570–1582, 2005.

[10] L. Zelnik-Manor, K. Rosenblum and Y. C. Eldar, Dictionary Optimization for Block-Sparse Representations, IEEE Trans. Signal Process., vol. 60, no. 4, pp. 2386–2395, 2012.

[11] J. M. Chambers, Partial Sorting, CACM, vol. 14, no. 5, pp. 357–358, 1971.

[12] C. Rusu and B. Dumitrescu, Stagewise K-SVD to design efficient dictionaries for sparse representations, IEEE Signal Processing Letters, vol. 19, no. 10, pp. 631–634, 2012.