A SEQUENTIAL DICTIONARY LEARNING ALGORITHM WITH ENFORCED SPARSITY

Abd-Krim Seghouane

Department of EEE
Melbourne School of Engineering
University of Melbourne, Australia

Muhammad Hanif

College of Engineering and Computer Science
The Australian National University
Canberra, Australia

ABSTRACT

Dictionary learning algorithms have received widespread acceptance when it comes to data analysis and signal representations problems. These algorithms alternate between two stages: the sparse coding stage and dictionary update stage. In all existing dictionary learning algorithms the use of sparsity has been limited to the sparse coding stage while presenting differences in the dictionary update stage which can be achieved sequentially or in parallel. The singular value decomposition (SVD) has been successfully used for sequential dictionary update. In this paper we propose a dictionary learning algorithm that include a sparsity constraint also in the dictionary update stage. The cost function used to include sparsity in the dictionary update stage is derived using the link between SVD and rank one matrix approximation. The effectiveness of the proposed dictionary learning method is tested on synthetic data and an image processing application. The results reveal that including a sparsity constraint in the dictionary update stage is not a bad idea.

***Index Terms*—** Dictionary learning, sparsity, SVD, sequential update, penalized rank one approximation.

1. INTRODUCTION

Dictionary learning methods have been successfully used in a number of signal and image processing applications among them image denoising [1][2], face recognition [3], compression [4] and fMRI data analysis [5][6][7]. These methods seek to uncover a linear multivariable latent structure in the observed data by imposing the reasonable constraint that the estimates of the observed variables are sparse linear functions of some unknown regressors (atoms of the dictionary). The other particularity of these methods is that this set of regressors or the dictionary is also tuned iteratively to find the optimal linear multivariable latent model.

Dictionary learning algorithms consist of two stages: a sparse coding stage and a dictionary update stage. In the first stage the dictionary is kept constant and the sparsity assumption is used to produce sparse linear approximations of the observed data. In the second stage, the coefficients of the linear combination are kept constant and the dictionary is updated to minimize a certain cost function. The dictionary learning methods iterate between these two stages until convergence. The performance of these methods strongly depend on the dictionary update stage since most of these methods share a similar sparse coding stage. Dictionary learning through the dictionary update stage can also be made sequential or parallel. For a specific strategy or goal (characterized by the cost function used to update the dictionary), the parallel approach will generally require a lower computational cost than the sequential approach that uses the same strategy. While the parallel update approach may be preferred for its advantage in computational complexity, the sequential approach generally offers better results because it generates finer tuned dictionary atoms.

Probabilistic and non-probabilistic approaches have been adopted for the derivation of dictionary learning algorithms. Most of the proposed algorithms have kept the two stages optimization procedure, the difference appearing mainly in the dictionary update stage. The maximum likelihood and *a posteriori* approaches have been used in [8][9][10][11] to derive an estimate of the dictionary in parallel. The square of the Frobenius norm was used in [1] to derive a sequential dictionary update stage. Other examples of sequential dictionary learning algorithms can be found in [12][13].

The sparsity constraint used in the sparse coding stage is the pillar of any dictionary learning algorithm. While this constraint is always used in the sparse coding stage, it has not been used in the dictionary update stage. Since there is no reason for not using a sparsity constraint also in the dictionary update stage, in this paper we revisit one of the most popular dictionary learning method [14], where we include the sparsity constraint also in the dictionary update stage. As revealed by the results obtained on different experiments, including the sparsity constraint also in the second stage of the algorithm appears to be a good idea. The contributions in this paper are therefore two: first we propose a new approach for dictionary learning where sparsity is also enforced in the dictionary update stage and second using the penalized matrix decomposition framework [15], we proposed an efficient approach for including the sparsity constraint in sequential dictionary learning.

2. BACKGROUND

Let $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N]$ where $\mathbf{y}_i \in \mathbf{R}^n$ be the set of signal to be represented. Given a dictionary $\mathbf{D} \in \mathbf{R}^{n \times K}$ containing a set of K regressors (dictionary atoms) $\mathbf{d}_k \in \mathbf{R}^n$, dictionary learning algorithms generate a representation of signal $\mathbf{y}$ as a sparse linear combination of the atoms $\mathbf{d}_k$ for $k = 1, ..., K$, $K << N$

$$\hat{\mathbf{y}}_i = \mathbf{D}\mathbf{x}_i$$

where $\mathbf{x}_i \in \mathbf{R}^K$ represent the corresponding signal strength. This signal is a sparse representation vector such that $||\mathbf{x}_i||_0 = s << K$ ($\| \cdot \|_0$ is the l_0 norm). Dictionary learning algorithms distinguish themselves from traditional model-based method by the fact that, in addition to $\mathbf{x}_i$, they also train the dictionary $\mathbf{D}$ to better fit the data set $\mathbf{Y}$. Given $\mathbf{Y}$, $\mathbf{D}$ is trained to minimize the error $||\mathbf{Y} - \mathbf{D}\mathbf{X}||_F^2$ where $\mathbf{X}$ are the sparse representations determined in the sparse coding stage. Dictionary learning algorithms generate a solution by iteratively alternating between the sparse coding stage

$$\hat{\mathbf{x}}_i = \arg\min_{x_i} \| \mathbf{y}_i - \mathbf{D}\mathbf{x}_i \|^2; \textit{ subject to } \| \mathbf{x}_i \|_0 \leq s \quad (1)$$

for $i = 1, ..., N$ and the dictionary update stage for the obtained $\mathbf{X}$ from the sparse coding stage

$$\mathbf{D} = \arg\min_{D} \| \mathbf{Y} - \mathbf{D}\mathbf{X} \|_F^2 . \quad (2)$$

Dictionary learning algorithms are often sensitive to the choice of s. Finding the optimal s corresponds to a problem of model order selection that can be resolved using a univariate linear model selection criterion [16, 17, 18, 19]. The update step can either be sequential as in [14] or in parallel as in [11]. In sequential dictionary learning, the dictionary update minimization problem (2) is split into K sequential minimizations, optimizing the cost function (2) for each atom individually while keeping the remaining atoms fixed. In the method proposed in [14], which has become a benchmark in dictionary learning, each column $\mathbf{d}_k$ of $\mathbf{D}$ and its corresponding row of coefficients $\mathbf{x}_k$ are updated based on a rank-1 matrix approximation of the error for all the signals when $\mathbf{d}_k$ is removed

$$\{\mathbf{d}_k, \mathbf{x}_k\} = \arg\min_{\mathbf{d}_k, \mathbf{x}_k^{row}} \|\mathbf{E}_k - \mathbf{d}_k\mathbf{x}_k^{row}\|_F^2. \quad (3)$$

where $\mathbf{E}_k = \mathbf{Y} - \sum_{i=1, i\neq k}^{K} \mathbf{d}_i\mathbf{x}_i^{row}$ The singular value decomposition (SVD) of $\mathbf{E}_k = \mathbf{U}\Delta\mathbf{V}^\top$ is used to find the closest rank-1 matrix approximation of $\mathbf{E}_k$ [20]. The $\mathbf{d}_k$ update is taken as the first column of $\mathbf{U}$ and the $\mathbf{x}_k^{row}$ update is taken as the first column of $\mathbf{V}$ multiplied by the first element of Δ, $\mathbf{x}_k^{row} = \lambda\widetilde{\mathbf{x}}_k^{row}$. The motivation of the proposed approach comes from the observation that the rank-1 approximation obtained using the SVD and written as $\mathbf{d}_k\mathbf{x}_k^{row}$ can also be obtained by alternating minimization [20] of

$$\|\mathbf{E}_k - \mathbf{d}_k\mathbf{x}_k^{row}\|_F^2 = \text{tr}\left\{(\mathbf{E}_k - \mathbf{d}_k\mathbf{x}_k^{row})(\mathbf{E}_k - \mathbf{d}_k\mathbf{x}_k^{row})^\top\right\}$$

$$= \|\mathbf{E}_k\|_F^2 - 2\mathbf{d}_k^\top\mathbf{E}_k\mathbf{x}_k^{row^\top} + \|\mathbf{d}_k\|^2.\|\mathbf{x}_k^{row}\|^2. \quad (4)$$

subject to $\|\mathbf{d}_k\|^2 = 1$, which gives

$$\mathbf{d}_k = \frac{\mathbf{E}_k\mathbf{x}_k^{row^\top}}{\| \mathbf{E}_k\mathbf{x}_k^{row^\top} \|_2} \text{ and } \mathbf{x}_k^{row} = \mathbf{d}_k^\top\mathbf{E}_k. \quad (5)$$

These equations can be used to justify the power algorithm, which, if initialized randomly, converges almost surely to a least square rank one fit. Using this observation a dictionary update can be obtained by iterating (5) until convergence or by applying only one iteration of the equations instead of the computationally expensive SVD.

3. MOTIVATION

The direct application of the SVD to $\mathbf{E}_k$ as described above to generate an update of $\mathbf{d}_k$ and $\mathbf{x}_k^{row}$ inevitably led to the loss of sparsity in $\mathbf{x}_k^{row}$ since the update vector may contain all nonzero entries. To remedy to this loss of sparsity problem, the algorithm proposed in [14] restricts the optimization of (3) only to the signals $\mathbf{y}_i$ that use the atom $\mathbf{d}_k$. Defining the index set $w_k = \{i | 1 \leq i \leq N; \mathbf{x}_k^{row}(i) \neq 0\}$, the SVD is applied on the matrix formed by the columns of $\mathbf{E}_k$ indexed by the elements of w_k. Denoting by $\mathbf{I}_{w_k}$ the $N \times |w_k|$ submatrix of the $N \times N$ identity matrix obtained by retaining only those columns whose index numbers are in w_k, we see that the restricted matrix $\mathbf{E}_k^R = \mathbf{E}_k\mathbf{I}_{w_k}$. Taking the SVD of $\mathbf{E}_k^R = \mathbf{U}\Delta\mathbf{V}^\top$ rather than that of $\mathbf{E}_k$ for dictionary update will only modify the the nonzero entries of $\mathbf{x}_k^{row}$.

As indicated in [14] not *remembering* the sparsity in the dictionary update stage may lead to a lower performance dictionary learning method. An alternative to this sparsity *remembering* approach is: rather than only updating the nonzero entries of $\mathbf{x}_k^{row}$ in the dictionary update stage we also propose to re-update the sparsity of $\mathbf{x}_k^{row}$. The resulting problem is a regularized rank one matrix approximation where the penalty is introduced in the minimization problem to promote sparsity of $\mathbf{x}_k^{row}$. Form the connection of principal component analysis (PCA) with SVD this can also be seen as a problem of sparse PCA [21][22] where the ℓ_1 penalty was used to promote sparsity of the loading vectors and improve the interpretability of PCA.

4. PROPOSED DICTIONARY UPDATE STAGE WITH ENFORCED SPARSITY

Given the above description, we seek an optimization framework to achieve a rank one approximation $\mathbf{d}_k\mathbf{x}_k^{row}$ with a sparse vector $\mathbf{x}_k^{row}$ where $\mathbf{x}_k^{row} = \lambda\widetilde{\mathbf{x}}_k^{row}$, λ being the largest singular value of $\mathbf{E}_k$. From the numerous proposals for sparse PCA we adopt the popular penalized regression approach [22]. With this approach, the updates of $\mathbf{d}_k$ and $\mathbf{x}_k^{row}$ are obtained by alternating minimization of

$$\{\mathbf{d}_k, \mathbf{x}_k^{row}\} = \arg\min_{\mathbf{d}_k, \mathbf{x}_k^{row}} \|\mathbf{E}_k - \mathbf{d}_k\mathbf{x}_k^{row}\|_F^2 + \alpha\|\mathbf{x}_k^{row}\|_1 \quad (6)$$

$$\text{subject to } \|\mathbf{d}_k\|_2 = 1$$

where α is a non-negative penalty parameter controlling the amount of sparsity in $\mathbf{x}_k^{row}$ (increasing α increases the amount of sparsity in $\mathbf{x}_k^{row}$) and the penalty encouraging sparsity is taken as the l_1-norm [23][24]. The use of $\mathbf{x}_k^{row}$ instead of $\widetilde{\mathbf{x}}_k^{row}$ in the l_1 penalty is justified by the fact that $\mathbf{x}_k^{row}$ is a unit vector and thus subject to scale constraint. This in turn will invalidate its use of the l_1 penalty. For fixed $\mathbf{d}_k$ and $\|\mathbf{d}_k\|_2 = 1$, the $\mathbf{x}_k^{row}$ that minimizes (6) is given by

$$\begin{aligned}\mathbf{x}_k^{row} &= \arg\min_{\mathbf{x}_k^{row}} \|\mathbf{E}_k\|_F^2 + \|\mathbf{x}_k^{row}\|^2 + \alpha\|\mathbf{x}_k^{row}\|_1 \\ &- 2\mathbf{d}_k^\top \mathbf{E}_k \mathbf{x}_k^{row^\top}. \end{aligned} \quad (7)$$

Hence the solution is given by

$$\mathbf{x}_k^{row} = \text{sgn}(\mathbf{d}_k^\top \mathbf{E}_k).\left(|\mathbf{d}_k^\top \mathbf{E}_k| - \frac{\alpha}{2}\mathbf{1}_{(N)}^\top\right)_+ \quad (8)$$

where $\mathbf{1}_{(N)}$ is a vector of ones of size N. For fixed $\mathbf{x}_k^{row}$, the $\mathbf{d}_k$ that minimizes (6) is derived from

$$\mathbf{d}_k = \arg\min_{\mathbf{d}_k} -2\mathbf{d}_k^\top \mathbf{E}_k \mathbf{x}_k^{row^\top} + \|\mathbf{d}_k\|^2.\|\mathbf{x}_k\|^2 \quad (9)$$

which with the constraint $\|\mathbf{d}_k\|_2 = 1$ gives

$$\mathbf{d}_k = \frac{\mathbf{E}_k \mathbf{x}_k^{row^\top}}{||\mathbf{E}_k \mathbf{x}_k^{row^\top}||_2}. \quad (10)$$

For a penalty of the form $\alpha^2 \parallel \mathbf{x}_k^{row} \parallel_0 = \alpha^2 \sum_{i=1}^N I(x_k^{row}(i) \neq 0)$; a measure that counts the number of nonzero coefficients; where $x_k^{row}(i)$ is the i^{th} entry of $\mathbf{x}_k^{row}$, instead of $\alpha\|\mathbf{x}_k^{row}\|_1$ in (6) the solution of (7) is given by $\mathbf{x}_k^{row} = \mathbf{I}\left(|\mathbf{d}_k^\top \mathbf{E}_k| > \alpha \mathbf{1}_{(N)}^\top\right)\mathbf{d}_k^\top \mathbf{E}_k$.

The resulting dictionary learning algorithm is depicted in table 1. Instead of the SVD of $\mathbf{E}_k^R$ [14], the updates of $\mathbf{d}_k$ and $\mathbf{x}_k^{row}$ are found by iterating (8) and (10) until convergence. This updating strategy is similar in spirit to using (5) for computing $\mathbf{d}_k$ and $\mathbf{x}_k^{row}$. The selection of the penalty parameter α can be obtained using a model selection criterion or cross validation. The simplified dictionary update stage used in this paper is obtained by applying a single iteration of (8) and (10) rather alternating until convergence. The computational cost of this iteration is $O(nN)$ compared to the $O(ln^2|w_k| + l'|w_k|^3)$ computational cost of an SVD [20] used in [14].

5. EXPERIMENTAL RESULTS

We compared the proposed algorithm with K-SVD [14] and MOD [11] on two applications: dictionary recovery and fill-in missing image pixels. In each case we used a normalized overcomplete dictionary, learned either from the observed noisy data or trained over some training set.

Table 1. Stepwise description of the proposed sequential dictionary learning algorithm with enforced sparsity

Given: $\mathbf{Y} \in \mathbf{R}^{n\times N}$, $\mathbf{D}_{ini}$, s, α, J.
Set $\mathbf{D} = \mathbf{D}_{ini}$
For *i=1 to J*
1: *Sparse Coding Stage:*
Find sparse coefficients X, by approximately solving
$\hat{\mathbf{x}}_i = \arg\min_{x_i} \parallel \mathbf{y}_i - \mathbf{D}\mathbf{x}_i \parallel^2$;
subject to $\parallel \mathbf{x}_i \parallel_0 \leq s \quad i = 1, ..., N$
2: Dictionary Update Stage:
For each column $k = 1, 2, ..., K$ in $\mathbf{D}$,
2.a: Compute the error matrix using
$\mathbf{E}_k = \mathbf{Y} - \sum_{i=1, i\neq k}^K \mathbf{d}_i \mathbf{x}_i^{row}$
2.b: Update the row $\mathbf{x}_k^{row}$ and its sparsity using
$\mathbf{x}_k^{row} = \text{sgn}(\mathbf{d}_k^\top \mathbf{E}_k).\left(|\mathbf{d}_k^\top \mathbf{E}_k| - \frac{\alpha}{2}\mathbf{1}_{(N)}^\top\right)_+$
2.c: Update the dictionary atom $\mathbf{d}_k$ using
$\mathbf{d}_k = \frac{\mathbf{E}_k \mathbf{x}_k^{row^\top}}{||\mathbf{E}_k \mathbf{x}_k^{row^\top}||_2}$
end.
Output: **D**,**X**

5.1. Synthetic data

In the first experimental test, similar to the reported works [14] and [11], the proposed method is evaluated with a synthetically generated signal where the learned dictionary is compared with the actual dictionary that generates the signal. This test will demonstrates the dictionary learning accuracy of each method. A generating dictionary, D_g, of size 20×100 is generated with i.i.d uniformly distributed entries. Each column (atom) of D_g is normalized to unit ℓ_2 norm. A training set Y of 1500 signals with dimensions 20 is generated. Each training signal is created by a linear combination of randomly located s atoms from D_g. Finally an equal white Gaussian noise is added to each training signal to maintain a uniform signal to noise ratio (SNR)$\in [15, 20, 30, 40]$ dB.

In each algorithm the learned dictionary, D_l, is initialized with a (same) set of training signals. In all cases the sparse coefficients are estimated using the orthogonal matching pursuit (OMP), with s coefficient approximation for each signal. The experiments are run for different sparsity levels, s = 2, 3, 4 and 5 and for each sparsity level the algorithms are iterated $j = 300$ times, where j indicates the iterations number after which the output is almost stable. For each sparsity level, we generate signals with SNR levels 15, 20, 30 and 40 dB. For each sparsity and SNR combination we run 10 trails and record the mean values.

The learned dictionaries by each algorithm are compared against the generating dictionary in the similar way as in [14][11]. The average numbers of restored dictionary atoms

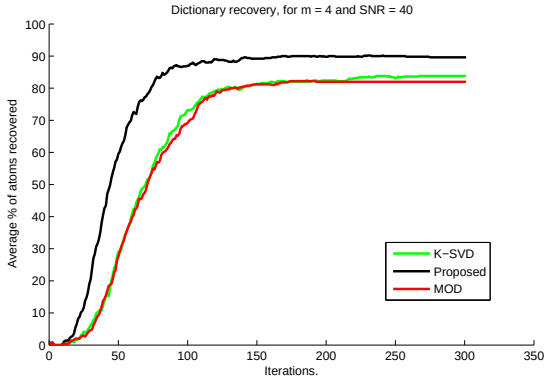

Fig. 1. Average number of dictionary atoms retrieved at each iteration.

Table 2. Average number of restored dictionary atoms.

	Sparsity (s)		**Sparsity (s)**	
	3	4	3	4
Method	**SNR 15 dB**		**SNR 30 dB**	
K-SVD	67.40	50.62	80.13	65.02
MOD	61.64	47.27	78.62	62.32
Proposed	78.68	73.15	90.48	87.53

for SNR 15 and 30 dB are compared in table 2, which conclude the performance edge of the proposed method over K-SVD and MOD algorithms. In figure 1 we plot the number of dictionary atoms retrieved at each iteration of the algorithms for s = 4 and 40 dB SNR. These results shows comparatively improved performance of our proposed method.

5.2. Fill-in missing image pixels

In our second experiment the learned dictionaries are tested for the estimation of missing image data, i-e filling the missing pixels in an image. A training data set $\mathbf{Y}$ is constructed by randomly selecting 2000 patches of size 8×8 from a set of training images[1]. We applied K-SVD, MOD, and the proposed method to learn dictionaries of size 64×100 from Y, with the same sparsity level and number of iterations, $j = 10$. We than select an input image from the set of training images. The image is divided into N non-overlapping patches of size 8×8, to form the image matrix $I \in \mathbf{R}^{64 \times N}$. In each image patch, I_i, a fraction of m random pixels are deleted, set to zero, where $m \in [0.2,\ 0.7]$. For each image patch with missing pixels, the sparse coefficients are estimated under the learned dictionaries using OMP. The estimated sparse coefficients vector of each patch is denoted by x_i, where i indicates the number of patch. The reconstructed patch is then obtained as, $\hat{\mathbf{I}}_i = D \cdot x_i$ where D is the learned dictionary.

[1]The training set consists of the well known Lena, Barbara, Cameraman, Peppers and Boat images of size 256×256.

Table 3. Fill-in missing pixels comparison in terms of SSD.

Method	**m**	
	0.3	0.5
K-SVD	16.21	22.01
MOD	16.80	22.53
Proposed	15.05	20.60

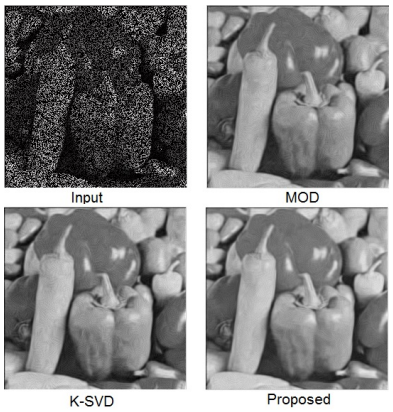

Fig. 2. Missing Pixel: Visual comparison.

In table 3 we present the result comparisons in terms of sum of squared difference (SSD), calculated from the reconstructed image and the original Lena image for $m = 0.3$ and 0.5. As can be seen, the proposed method produce better quality estimation compared to K-SVD and MOD methods. Moreover, the computational cost of the proposed method is less expensive than K-SVD which uses $K-$times SVD to sequentially update the dictionary. A visual comparison for $m = 0.6$ is presented in figure 2, where a corrupted image is reconstructed using the learned dictionaries.

6. CONCLUSION

While the sparsity constraint constitutes a primary ingredient of any dictionary learning method, it has only been included in the sparse coding stage. In this paper a new sequential dictionary learning algorithm has been proposed by efficiently including the sparsity constraint also in the dictionary update stage. Compared to state of the art methods, the proposed algorithm is computationally more efficient and generates better results.

7. REFERENCES

[1] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over leanred dictionaries," *IEEE Transactions on Image Processing*, vol. 15, pp. 3736–3745, 2006.

[2] A. K. Seghouane and M. Hanif, "A basis expansion based regularized sequential dictionary learning algorithm for image restoration," *IEEE International Conference on Image Processing (ICIP)*, 2015.

[3] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, 2008.

[4] J. Zepeda, C. Guillemot, and E. Kijak, "Image compression using the iteration-tuned and aligned dictionary," *In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 793–796, 2011.

[5] M. U. Khalid and A. K. Seghouane, "A single SVD sparse dictionary learning algorithm for fMRI data analysis," *In Proceedings of IEEE International Workshop on Statistical signal Processing*, pp. 65–68, 2014.

[6] M. U. Khalid and A. K. Seghouane, "Improving functional connectivity detection in fMRI by combining sparse dictionary learning and canonical correlation analysis," *In Proceedings of IEEE International Symposium on Biomedical Imaging*, pp. 286–289, 2013.

[7] M. U. Khalid and A. K. Seghouane, "Constrained maximum likelihood based efficient dictionary learning for fMRI analysis," *In Proceedings of IEEE International Symposium on Biomedical Imaging*, pp. 45–48, 2014.

[8] M. Hanif and A. K. Seghouane, "Maximum likelihood orthogonal dictionary learning," *IEEE Workshop on Statistical Signal Processing (SSP)*, pp. 1–4, 2014.

[9] K. Kreutz-Delgado et, J. F. Murray, B. D. Rao, K. Engan, T. W. Lee, and T. J. Sejnowski, "Dictionary learning algorithms for sparse representation," *Neural Computation*, vol. 15, pp. 349–396, 2003.

[10] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Computation*, vol. 12, pp. 337–365, 2000.

[11] K. Engan, S. O. Aase, and J. Hakon-Husoy, "Method of optimal directions for frame design," *IEEE Int. Conference on Acoustics, Speech, and Signal Processing*, pp. 2443–2446, 1999.

[12] S. K. Sahoo and A. Makur, "Dictionary training for sparse representation as generalization of K-means clustering," *IEEE Signal Processing Letters*, vol. 20, pp. 587–590, 2013.

[13] Z. Jiang, Z. Lin, and L. Davis, "Label consistent K-SVD: Learning a discriminative dictionary for recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 2651–2664, 2013.

[14] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for desiging overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, pp. 4311–4322, 2006.

[15] D. M. Witten, R. Tibshirani, and T. Hastie, "A penalized matrix decompostion, with applications to sparse principal componnents and canonical correlation analysis," *Biostatistics*, vol. 10, pp. 515–534, 2009.

[16] A. K. Seghouane and M. Bekara, "A small sample model selection criterion based on the kullback symmetric divergence," *IEEE Transactions on Signal Processing*, vol. 52, pp. 3314–3323, 2004.

[17] A. K. Seghouane and S. I. Amari, "The AIC criterion and symmetrizing the kullback-leibler divergence," *IEEE Transactions on Neural Networks*, vol. 18, pp. 97–106, 2007.

[18] A. K. Seghouane, "Asymptotic bootstrap corrections of AIC for linear regression models," *Signal Processing*, vol. 90, pp. 217–224, 2010.

[19] A. K. Seghouane and A. Cichocki, "Bayesian estimation of the number of principal components," *Signal Processing*, vol. 87, pp. 562–568, 2007.

[20] G. H. Golub and C. f. Van Loan, *Matrix Computations*, Johns Hopkins, 1996.

[21] I. T. Jolliffe, N. T. Trendafilov, and M. Uddin, "A modified principal component technique based on the LASSO," *Journal of Computational Graphical Statistics*, vol. 12, pp. 531–547, 2003.

[22] H. Shen and J. Z. Huang, "Sparse principal component analysis via regularized low rank matrix approximation," *Journal of Multivariate Analysis*, vol. 99, pp. 1015–1034, 2008.

[23] A. K. Seghouane, "Model selection criteria for image restoration," *IEEE Transactions on Neural Networks*, vol. 20, pp. 1357–1363, 2009.

[24] A. K. Seghouane, "A note on image restoration using Cp and MSE," *IEEE signal Processing Letters*, vol. 15, pp. 61–64, 2008.