SUBSET PURSUIT FOR ANALYSIS DICTIONARY LEARNING

Ye Zhang[1,2]*, Haolong Wang*[1]*, Tenglong Yu*[1]*, Wenwu Wang*[2]

[1]Department of Electronic and Information Engineering, Nanchang University, Nanchang, China
[2]Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, United Kingdom
zhangye@ncu.edu.cn, w.wang@surrey.ac.uk, 710215399@qq.com, 910140889@qq.com

ABSTRACT

Most existing analysis dictionary learning (ADL) algorithms, such as the Analysis K-SVD, assume that the original signals are known or can be correctly estimated. Usually the signals are unknown and need to be estimated from its noisy version with some computational efforts. When the noise level is high, estimation of the signals becomes unreliable. In this paper, a simple but effective ADL algorithm is proposed, where we directly employ the observed data to compute the approximate analysis sparse representation of the original signals. This eliminates the need for estimating the original signals as otherwise required in the Analysis K-SVD. The analysis sparse representation can be exploited to assign the observed data into multiple subsets, which are then used for updating the analysis dictionary. Experiments on synthetic data and natural image denoising demonstrate its advantage over the baseline algorithm, Analysis K-SVD.

Index Terms— Analysis sparse representation; dictionary learning; cosparse model; image denoising

1. INTRODUCTION

Modeling signals as sparse linear combinations of a few atoms selected from a learned dictionary has been the focus of much recent research in many signal processing fields such as image denoising, audio processing, compression, and more. A popular model for sparse representation is the synthesis model. Consider a signal $\mathbf{x} \in R^M$, the synthesis sparse representations of $\mathbf{x}$ over a dictionary $\mathbf{D}$ can be described as $\mathbf{x} = \mathbf{Da}$, where $\mathbf{D} \in R^{M\times N}$ is a possibly overcomplete dictionary ($N \geq M$), and $\mathbf{a} \in R^N$, containing the coding coefficients, is assumed to be sparse, i.e. $\|\mathbf{a}\|_0 = k \ll N$. The model assumes that the signal $\mathbf{x} \in R^M$ can be described as a linear combination of only a few columns (i.e. signal atoms) from the dictionary $\mathbf{D}$. Thus, the performance of the model hinges on the representation of the signals with an appropriate dictionary. In the past decade, a great deal of effort has been dedicated to learning the dictionary $\mathbf{D}$ from signal examples [1, 2, 3].

Recently, an alternative sparse model called analysis sparse model was proposed in [4, 5, 6, 7]. In this model, an overcomplete analysis dictionary or analysis operator $\mathbf{\Omega} \in R^{P\times M}$ ($P \geq M$) is sought to transform the signal vector $\mathbf{x} \in R^N$ to a high dimensional space, i.e. $\mathbf{\Omega x} = \mathbf{z}$, where the analysis coefficient vector $\mathbf{z} \in R^P$ is called the analysis representation of $\mathbf{x}$ and assumed to be sparse. In this model, the signal $\mathbf{x}$ is characterized by the location of the zero entries of $\mathbf{z}$. In other words, the rows of $\mathbf{\Omega}$ that are associated with zero entries in $\mathbf{z}$ define a subspace that the signal $\mathbf{x}$ belongs to, as opposed to the few non-zero entries of $\mathbf{a}$ in the synthesis model. The dictionary $\mathbf{\Omega}$ is often learned from the observed signals $\mathbf{Y} = [\mathbf{y}_1\ \mathbf{y}_2\ ...\ \mathbf{y}_K] \in R^{M\times K}$ measured in the presence of additive noise, i.e. $\mathbf{Y} = \mathbf{X} + \mathbf{V}$, where $\mathbf{X} = [\mathbf{x}_1\ \mathbf{x}_2\ ...\ \mathbf{x}_K] \in R^{M\times K}$ contains the original signals, $\mathbf{V} = [\mathbf{v}_1\ \mathbf{v}_2\ ...\ \mathbf{v}_K] \in R^{M\times K}$ is noise and K is the number of signals. Compared to the extensive study for synthesis dictionary learning, however, the analysis dictionary learning problem has received much less attention with only a few algorithms proposed recently [5, 7, 8].

In [5], an ℓ_1-norm penalty function is applied to the representation $\mathbf{\Omega x}_i$, and a projected subgradients algorithm is proposed for analysis operator learning. This work employs a uniformly normalized tight frame as a constraint on the dictionary to avoid the trivial solution. However, the method adds a rather arbitrary constraint for the learning problem, and this constraint limits the possible $\mathbf{\Omega}$ to be learned. Exploiting the fact that a row of the analysis dictionary $\mathbf{\Omega}$ is orthogonal to a sub-set of training signals $\mathbf{X}$, a sequential minimal eigenvalue based ADL algorithm was proposed in [8]. Once the sub-set is found, the corresponding row of the dictionary can be updated with the eigenvector associated with the smallest eigenvalue of the autocorrelation matrix of these signals. However, as the number of the rows in $\mathbf{\Omega}$ increases, so does the computational cost of the method. In [7], the Analysis K-SVD algorithm is proposed for analysis dictionary learning. By keeping $\mathbf{\Omega}$ fixed, the optimal backward greedy algorithm was employed to estimate a sub-matrix of $\mathbf{\Omega}$ whose rows are orthogonal to $\hat{\mathbf{X}}$, i.e. the estimate of $\mathbf{X}$ from $\mathbf{Y}$. A data set,

This work was supported by the National Natural Science Foundation of China (Grant Nos. 61162014, 61210306074), the Natural Science Foundation of Jiangxi (Grant No. 20122BAB201025) and was supported in part by the Engineering and Physical Sciences Research Council (EPSRC) of the UK (Grant Nos. EP/H050000/1 and EP/ H012842/1).

i.e. the sub-matrix of $\mathbf{Y}$, can be obtained, with its columns corresponding to that of $\hat{\mathbf{X}}$. The smallest singular values of this sub-matrix are then used to update $\mathbf{\Omega}$. The algorithm is effective, but it needs to pre-estimate the signal $\mathbf{x}$ in order to learn the dictionary.

The ADL algorithms mentioned above, all assume that the signals $\mathbf{X}$ are known or can be accurately estimated from its noisy version $\mathbf{Y}$. However, in practice, the signals $\mathbf{X}$ are unknown or need to be estimated from $\mathbf{Y}$ with some computational efforts. When the noise level is high, estimate of $\mathbf{X}$ becomes unreliable. In this paper, using the cosparsity of $\mathbf{x}$ (with respect to $\mathbf{\Omega}$), we propose a simple but effective ADL algorithm. In this algorithm, we directly use the observed data $\mathbf{Y}$ for learning the dictionary, i.e. without having to pre-estimate $\mathbf{X}$ (as done in [7]). Simulation results show that the denosing performance of our algorithm is better than that of the Analysis K-SVD, especially when the noise level in the observed signals increases. In addition, the proposed algorithm is much faster than the Analysis K-SVD algorithm.

The paper is organized as follows. In Section 2, we discuss the cosparse analysis model and briefly describe the Analysis K-SVD algorithm. In Section 3, we describe the proposed Subset Pursuit Analysis Dictionary Learning algorithm (SP-ADL). In Section 4, we show some experimental results, before concluding the paper in Section 5.

2. THE COSPARSE ANALYSIS MODEL

The cosparse analysis model can be described as follows: for a signal $\mathbf{x} \in R^M$ and a fixed redundant analysis dictionary $\mathbf{\Omega} \in R^{P\times M}(P > M)$, the cosparsity l of the cosparse analysis model is

$$l = P - \|\mathbf{\Omega}\mathbf{x}\|_0 \tag{1}$$

where the ℓ_0 quasi-norm $\|\cdot\|_0$ counts the number of nonzero components in its argument. The quantity l denotes the number of zeros in the vector $\mathbf{\Omega}\mathbf{x}$, which implies that l rows in $\mathbf{\Omega}$ are orthogonal to the signal $\mathbf{x}$, and these rows define the cosupport Λ, i.e. $\mathbf{\Omega}_\Lambda\mathbf{x} = 0$, where $\mathbf{\Omega}_\Lambda$ is a sub-matrix of $\mathbf{\Omega}$ that contains the rows from $\mathbf{\Omega}$ indexed by Λ. In this case the signal $\mathbf{x}$ is said to be l-cosparse and characterized by its cosupport Λ. It is clear that the dimension of the subspace that signal $\mathbf{x}$ resides in is $r = M - l$. One can observe that the larger the l, the more cosparse the $\mathbf{x}$.

In the analysis model, if the true cosupport Λ is known, the signal $\mathbf{x}$ can be recovered from its noisy version $\mathbf{y} = \mathbf{x}+\mathbf{v}$ by [7]

$$\hat{\mathbf{x}} = \left(\mathbf{I} - \mathbf{\Omega}_\Lambda^\dagger\mathbf{\Omega}_\Lambda\right)\mathbf{y} \tag{2}$$

However, in general the dictionary $\mathbf{\Omega}$ and the cosupport Λ are unknown. To find $\mathbf{\Omega}$ and Λ, the Analysis K-SVD algorithm in [7] assumes that every example of the observation set $\mathbf{Y}$ is a noisy version of the signal residing in an r-dimensional subspace and all examples have the same co-rank of $M - r$ related to the dictionary $\mathbf{\Omega}$, and then the optimization task can be described as follows:

$$\begin{aligned}&\left(\hat{\mathbf{\Omega}}, \hat{\mathbf{X}}, \left(\hat{\Lambda}_i\right)_{i=1}^K\right) = \underset{\mathbf{\Omega},\mathbf{X},(\Lambda_i)_{i=1}^K}{\operatorname{argmin}} \|\mathbf{X} - \mathbf{Y}\|_F^2 \ \text{ s.t.} \\ &\mathbf{\Omega}_{\Lambda_i}\mathbf{x}_i = 0, \ 1 \le \forall i \le K \\ &Rank(\mathbf{\Omega}_{\Lambda_i}) = M - r, \ 1 \le \forall i \le K \\ &\left\|\mathbf{w}_j^T\right\|_2 = 1, \ 1 \le \forall j \le P\end{aligned} \tag{3}$$

where $\mathbf{x}_i$ is the i-th column of $\mathbf{X}$, Λ_i is the cosupport of $\mathbf{x}_i$, and $\mathbf{w}_j^T$ denotes the rows of $\mathbf{\Omega}$. In [7] a two-phase block-coordinate-relaxation approach is used for the optimization task. In the first phase, by keeping $\mathbf{\Omega}$ fixed, the backward greedy algorithm is employed to find the cosupport Λ by selecting the rows from $\mathbf{\Omega}$ one-by-one, and then the estimation of $\mathbf{X}$ is obtained. In the second phase, the estimation $\hat{\mathbf{X}}$ is assigned into sub-matrix $\hat{\mathbf{X}}_j \subseteq \hat{\mathbf{X}}$ whose columns are orthogonal to $\mathbf{w}_j$, and then a corresponding sub-matrix of $\mathbf{Y}$, i.e. $\mathbf{Y}_j \subseteq \mathbf{Y}$, is built and the singular vector corresponding to the smallest singular value of $\mathbf{Y}_j$ is used to update the $\mathbf{w}_j$. However, in the algorithm, the original signals $\mathbf{X}$ need to be estimated first before the dictionary can be updated. Due to use of greedy-like algorithms, estimation of $\mathbf{X}$ is computationally slow and also becomes unreliable with the increase of noise in $\mathbf{Y}$.

3. PROPOSED SUBSET PURSUIT ALGORITHM

In this section we present a subset pursuit algorithm for analysis dictionary learning, by directly using the observed data for learning the analysis dictionary, without having to pre-estimate $\mathbf{X}$ (as done in [7]). More specifically, we exploit the analysis representation of $\mathbf{Y}$ to obtain subset $\mathbf{Y}_j$, rather than using $\hat{\mathbf{X}}_j$ to determine $\mathbf{Y}_j$. As a result, the proposed algorithm is much faster than the Analysis K-SVD algorithm, as shown in the simulation section. In addition, the proposed algorithm offers better performance for image denoising than the Analysis K-SVD algorithm. In our method, we also assume that $\mathbf{X}$ has the same co-rank $M - r$ related to the dictionary $\mathbf{\Omega}$ as in Analysis K-SVD.

3.1. The proposed ADL model

Suppose we measure a signal of the form

$$\mathbf{y}_i = \mathbf{x}_i + \mathbf{v}_i \ \ i = 1, 2, ..., K \tag{4}$$

where $\mathbf{v}_i$ is the noise vector with a bounded ℓ_2 norm, say $\|\mathbf{v}_i\|_2 \le \sigma$, where σ denotes the noise level. According to (1), the task of ADL can be formed as

$$\min \|\mathbf{\Omega}\mathbf{x}_i\|_0 \tag{5}$$

To solve this problem is NP-complete [9]. Just like in the synthesis case, one might replace the ℓ_0 quasi-norm with the ℓ_1 norm

$$\min \|\mathbf{\Omega}\mathbf{x}_i\|_1 \tag{6}$$

where $\|\cdot\|_1$ is the ℓ_1 norm that sums the absolute values of a vector. In general, when the noise $\mathbf{v}_i$ is stationary and bounded, the cosparse model (6) has an approximate cosparse model

$$\min \|\mathbf{\Omega}\mathbf{y}_i\|_1 \tag{7}$$

This is the model that we consider in our work, as opposed to the model used in the work of [7].

3.2. Subset pursuit algorithm

Using (7), the analysis cosparse model is therefore written as

$$\mathbf{z}_i = \mathbf{\Omega}\mathbf{y}_i \tag{8}$$

where $\mathbf{z}_i = [z_{1i} \;\; z_{2i} \;\; \cdots \;\; z_{Pi}]^T \in R^{P\times 1}$ is the analysis representation of $\mathbf{y}_i$. Considering a single row $\mathbf{w}_j^T$ in the analysis dictionary $\mathbf{\Omega}$, (8) can be rewritten as

$$z_{ji} = \mathbf{w}_j^T\mathbf{y}_i \tag{9}$$

Then the absolute values of z_{ji} is

$$|z_{ji}| = \left|\mathbf{w}_j^T\mathbf{y}_i\right| = \left|\mathbf{w}_j^T\mathbf{x}_i + \mathbf{w}_j^T\mathbf{v}_i\right| \tag{10}$$

We know that $\left|\mathbf{w}_j^T\mathbf{x}_i + \mathbf{w}_j^T\mathbf{v}_i\right| \leq \left|\mathbf{w}_j^T\mathbf{x}_i\right| + \left|\mathbf{w}_j^T\mathbf{v}_i\right|$. Thus, in general the absolute values of z_{ji} has a small value when $\mathbf{w}_j$ is orthogonal to the signal $\mathbf{x}_i$, i.e. $\mathbf{w}_j^T\mathbf{x}_i = 0$. Thus, we can compute the analysis sparse representation of the observed data $\mathbf{y}_i$ to find whether $\mathbf{w}_j^T$ is orthogonal with the signal $\mathbf{x}_i$, rather than using $\hat{\mathbf{x}}_i$ which otherwise has to be estimated from $\mathbf{Y}$.

Because the co-rank is assumed to be $M - r$, which implies that $M - r$ rows in $\mathbf{\Omega}$ may be orthogonal to the data $\mathbf{y}_i$, we can regard $M - r$ smallest values in $\mathbf{\Omega}\mathbf{y}_i$ as zeros. The $\Lambda_i := \{j \mid \left|\mathbf{w}_j^T\mathbf{y}_i\right| \approx 0\}$, which is the co-support of $\mathbf{y}_i$, can be obtained by the locations of the zero entries in $\mathbf{\Omega}\mathbf{y}_i$. We can then assign $\mathbf{y}_i$ into the sub-set $\mathbf{Y}_j, \forall j \in \Lambda_i$. After the $\mathbf{Y}_j$ is found, the $\mathbf{w}_j$ is updated as follow [7]:

$$\hat{\mathbf{w}}_j = \arg\min_{\mathbf{w}_j} \left\|\mathbf{w}_j^T\mathbf{Y}_j\right\|_2^2 \;\text{ s.t. }\; \|\mathbf{w}_j\|_2 = 1 \tag{11}$$

For the optimization problem, the $\mathbf{w}_j$ can be updated using the eigenvector associated with the smallest eigenvalue of $\mathbf{Y}_j\mathbf{Y}_j^T$. This algorithm is described in the **Algorithm** table.

4. COMPUTER SIMULATION

To validate the proposed algorithm, we present results of two experiments. In the first experiment we show the performance of the proposed algorithm for synthetic dictionary recovery problems. The second experiment considers the natural images denoising problem. In these experiments, $\mathbf{\Omega}_0 \in R^{P\times M}$ is randomly generated and the each row of the $\mathbf{\Omega}_0$ is normalized.

Algorithm: SP-ADL

Input: Observed data $\mathbf{Y} \in R^{M\times K}$, the initial dictionary $\mathbf{\Omega}_0 \in R^{P\times M}$, the co-rank $M - r$ and the number of iterations T

Output: Dictionary $\mathbf{\Omega}$

Initialization: Set $\mathbf{\Omega} := \mathbf{\Omega}_0$, Let $\mathbf{Y}'$ be the column-normalised version of $\mathbf{Y}$, where $\mathbf{Y}' = [\mathbf{y'}_1 ... \mathbf{y'}_K] \in R^{M\times K}$

For $t = 1...T$ **do**

  For $i = 1...K$ **do**

- Compute $\mathbf{z}_i = \mathbf{\Omega}\mathbf{y'}_i$, select $M - r$ numbers of $|z_{ji}|$ which have the smallest values and find the cosupport Λ_i
- Assign corresponding $\mathbf{y}_i$ into $\mathbf{Y}_j, \forall j \in \Lambda_i$

  End for

  For $j = 1...P$ **do**

  Update $\mathbf{w}_j$:

  $\hat{\mathbf{w}}_j = \arg\min_{\mathbf{w}_j} \left\|\mathbf{w}_j^T\mathbf{Y}_j\right\|_2^2 \;\text{ s.t. }\; \|\mathbf{w}_j\|_2 = 1$

  End for

End for

4.1. Experiments on synthetic data

In the experiment of this subsection we use the proposed method to recover a dictionary that was used to produce the set of training data. We used the same experimental protocol as in [7]. In [7] the analysis dictionary $\mathbf{\Omega} \in R^{50\times 25}$ was generated with random Gaussian entries, and the data set consists of $K = 50000$ analysis signals each residing in a 4-dimensional subspace with both the noise-free setup and a noise setup ($\sigma = 0.04, SNR = 25dB$). If $\min_i(1 - \left|\hat{\mathbf{w}}_i^T\mathbf{w}_j\right|) < 0.01$, a row $\mathbf{w}_j^T$ in the true dictionary $\mathbf{\Omega}$ is regarded as recovered, where $\hat{\mathbf{w}}_i^T$ are the atoms of the trained dictionary. The results are presented in Figure 1. It can be observed that, after running the SP-ADL algorithm for 300 iterations, 90% of the rows in the true dictionary $\mathbf{\Omega}$ were reconstructed for the noise-free case and 84% for the noise one. In contrast, as shown in Figure 2, the Analysis K-SVD algorithm recovers 90% of the rows in the true dictionary $\mathbf{\Omega}$ for the noise-free setup and 88% for the noise case[1] after 100 iterations. Even though the Analysis K-SVD algorithm took fewer iterations to reach a similar recovery percentage, the running time in each iteration of the Analysis K-SVD algorithm is significantly higher than that in our proposed SP-ADL algorithm. The total runtime of our algorithm (for generating the results in Figure 1) is about 3125 and 3238 seconds for the noise-free and noise case, respectively. In contrast, the Analysis K-SVD took about 11420 or 11502

[1] In [7] the experimental results show 94% and 86%, respectively.

seconds respectively (Computer OS: Windows 7, CPU: Intel Core i5-3210M @ 2.50GHz, RAM: 4G). The main reason is because our algorithm does not need to estimate $\mathbf{X}$ in each iteration of the learning algorithm, as opposed to the Analysis K-SVD algorithm.

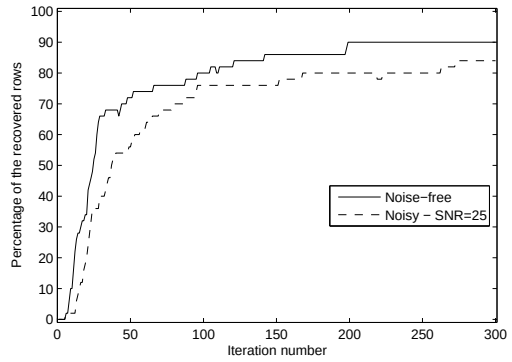


Fig. 1: Results of SP-ADL for synthetic data.

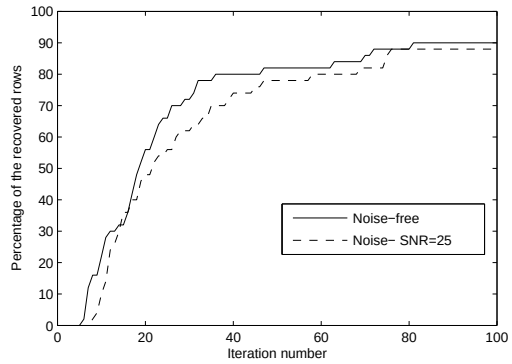


Fig. 2: Results of Analysis K-SVD for synthetic data.

4.2. Experiments on natural image denoising

In this experiment, using the test set consisting of three images commonly used in denoising (Lena, house and peppers) [7, 10], we perform denoising experiments and compare the performance of the SP-ADL algorithm with that of the Analysis K-SVD on the same experimental protocol. The denoising performance is evaluated by the peak signal to noise ratio (PSNR) defined as

$$\text{PSNR} = 20\log_{10}\frac{255}{\sigma_e} \tag{12}$$

where σ_e is the standard deviation of the pixelwise image error. The dictionary of size 63×49 is created by using a training set of size 20,000 of 7×7 image patches. Noise with different noise level σ, varying from 5 to 20 is added to these image patches. We also assume that the dimension of the subspace $r = 7$, the same as in [7]. In this experiment we apply 50 iterations to learn the analysis dictionary for both algorithms. The examples of the learned dictionaries are shown in Figure 3.

The learned dictionaries are employed for patch-based image denoising by using (2), the results, averaged over 5 trials, are presented in Table 1, from which we can observe that the performance of the SP-ADL algorithm is better than that of the Analysis K-SVD when the noise level is increased. In the same simulation environment, when $\sigma = 5$, the SP-ADL algorithm took about 477 (Lena), 479 (House) and 477 (Peppers) seconds to learn the dictionary, respectively, while the Analysis K-SVD took about 7887 (Lena), 7973 (House) and 8029 (Peppers) seconds, respectivly. Clearly, SP-ADL is much faster than the Analysis K-SVD.

Table 1: Image denoising results (PSNR in dB)

σ	Noisy	Denoising method	Lena	House	Peppers
5	34.15	Analysis K-SVD	38.43	39.20	37.89
		SP-ADL	38.40	38.90	37.73
10	28.13	Analysis K-SVD	34.85	35.27	33.80
		SP-ADL	35.13	35.23	33.83
15	24.61	Analysis K-SVD	32.59	33.00	31.31
		SP-ADL	33.22	33.25	31.61
20	22.11	Analysis K-SVD	31.42	31.49	29.80
		SP-ADL	31.90	31.88	30.05

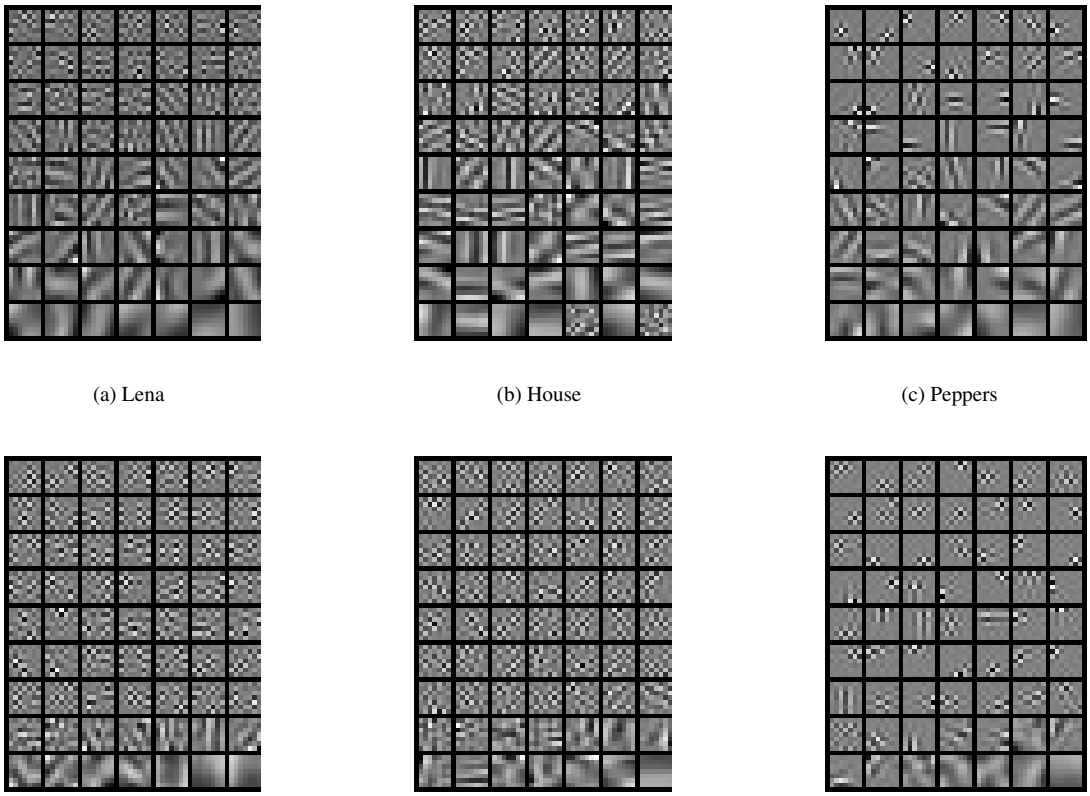
5. CONCLUSION

We have presented a simple and effective algorithm for analysis dictionary learning. It is closely related to the Analysis K-SVD algorithm. The difference between the two ADL algorithms is that in the SP-ADL algorithm we directly exploit the observed data for learning the analysis dictionary, without pre-estimating the signal $\mathbf{X}$ from its noisy version $\mathbf{Y}$, as opposed to the strategy taken by the Analysis K-SVD algorithm. The experiments performed have shown that SP-ADL algorithm is well suited for the analysis dictionary learning problem. It is easy to implement and computationally more efficient than the Analysis K-SVD algorithm.

6. REFERENCES

[1] K. Engan, S. O. Aase, and J. H. Husoy, "Method of optimal directions for frame design," in *Proc. Acoustics, Speech, and Signal Processing, IEEE International Conference on*, 1999, vol. 5, pp. 2443 –2446.

[2] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.



(a) Lena (b) House (c) Peppers

(d) Lena (e) House (f) Peppers

Fig. 3: The learned dictionaries of size 63×49 by using the SP-ADL and the Analysis K-SVD algorithms on the three images with noise level $\sigma = 5$. The results of the Analysis K-SVD are shown in (a) Lena, (b) House and (c) Peppers, with the others showing the results of SP-ADL.

[3] R. Rubinstein, A. M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045 –1057, 2010.

[4] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, "Cosparse analysis modeling - uniqueness and algorithms," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2011*, 2011, pp. 5804 –5807.

[5] M. Yaghoobi, S. Nam, R. Gribonval, and M. E. Davies, "Analysis operator learning for overcomplete cosparse representations," in *European Signal Processing Conference (EUSIPCO'11)*, Barcelona, Spain, 2011.

[6] G. Peyré and J. Fadili, "Learning analysis sparsity priors," in *Proc. of Sampta'11*, 2011.

[7] R. Rubinstein, T. Peleg, and M. Elad, "Analysis k-svd: A dictionary-learning algorithm for the analysis sparse model," *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 661 –677, 2013.

[8] B. Ophir, M. Elad, N. Bertin, and M. D. Plumbley, "Sequential minimal eigenvalues: An approach to analysis dictionary learning," in *In Proc 19th European Signal Processing Conference*, 2011, pp. 1465–1469.

[9] J. A. Tropp, "Just relax: convex programming methods for identifying sparse signals in noise," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1030 –1051, march 2006.

[10] S. Roth and M. J. Black, "Fields of experts," *Int. J. Comput. Vision*, vol. 82, no. 2, pp. 205–229, Apr. 2009.