K-PLANE CLUSTERING ALGORITHM FOR ANALYSIS DICTIONARY LEARNING

Ye Zhang[1,2]*, Haolong Wang*[1]*, Wenwu Wang*[2]*, and Saeid Sanei*[3]

[1]Department of Electronic and Information Engineering, Nanchang University, Nanchang, China
[2]Department of Electronic Engineering, University of Surrey, Guildford, United Kingdom
[3]Departme of Computing, University of Surrey, Guildford, United Kingdom
Emails: zhangye@ncu.edu.cn, 710215399@qq.com, w.wang@surrey.ac.uk, s.sanei@surrey.ac.uk

ABSTRACT

Analysis dictionary learning (ADL) aims to adapt dictionaries from training data based on an analysis sparse representation model. In a recent work, we have shown that, to obtain the analysis dictionary, one could optimise an objective function defined directly on the noisy signal, instead of on the estimated version of the clean signal as adopted in analysis K-SVD. Following this strategy, a new ADL algorithm using K-plane clustering is proposed in this paper, which is based on the observation that, the observed data are co-planer in the analysis sparse model. In other words, the columns of the observed data form multi-dimensional subspaces (hyper-planes), and the rows of the analysis dictionary are the normal vectors of the hyper-planes. The normal directions of the K-dimensional concentration hyper-planes can be estimated using the K-plane clustering algorithm, and then the rows of the analysis dictionary which are the normal vectors of the hyper-planes can be obtained. Experiments on natural image denoising demonstrate that the K-plane clustering algorithm provides comparable performance to the baseline algorithms, i.e. the analysis K-SVD and the subset pursuit based ADL.

Index Terms— K-plane clustering; Analysis dictionary learning; Co-sparse; Image denoising

1. INTRODUCTION

Sparse representation has become a well-known topic in a wide range of fields, such as image processing and compressed sensing. Sparse representation is often built on a generative model where the signals of interest are described as a linear combination of a few atoms chosen from a redundant and predefined (or learned) dictionary. In this model, a signal $\mathbf{x} \in R^M$ is represented as $\mathbf{x} = \mathbf{D}\mathbf{a}$, where $\mathbf{D} \in R^{M\times N}$ is a possibly overcomplete dictionary ($N \geq M$), and $\mathbf{a} \in R^N$ is the coefficient vector, which is assumed to be sparse, i.e., $\|\mathbf{a}\|_0 = k \ll N$, where the ℓ_0 quasi-norm $\|\cdot\|_0$ counts the number of nonzero components in its argument. This signal model is typically referred to as the synthesis model. In the past decade, the synthesis model has been intensively studied [1, 2, 3], where the dictionaries used to fit the model, can be obtained by either selecting pre-defined transforms, or adapting the atoms from a set of training signals. Recent studies have shown that the dictionaries learned from training data may perform better than pre-defined transforms despite the fact that a higher computational cost may be involved.

Recently, an alternative sparse model called analysis sparse model has been proposed, and there are only a few algorithms proposed to learn the analysis dictionary [4, 5, 6, 7]. In analysis sparse model, the signal $\mathbf{x} \in R^M$ is assumed to be composed as $\mathbf{\Omega x} = \mathbf{z}$, where $\mathbf{\Omega} \in R^{P\times M}$ ($P \geq M$) is an overcomplete analysis dictionary, and $\mathbf{z} \in R^P$ is the analysis representation of $\mathbf{x}$, which is assumed to be sparse, and the number of non-zeros in $\mathbf{z} \in R^P$ is assumed to be much smaller than P, i.e. $\|\mathbf{z}\|_0 = k \ll P$. The location of the zero entries of $\mathbf{z}$ can be employed to describe the subspace that the signal $\mathbf{x}$ belongs to. In general, the dictionary $\mathbf{\Omega} \in R^{P\times M}$ ($P \geq M$) can be learned from the observed signals.

In [5], a projected subgradient algorithm is proposed for analysis operator learning. In this work, an ℓ_1-norm penalty function is applied to $\mathbf{\Omega x}$, and the algorithm employs a uniformly normalized tight frame as a constraint on the dictionary to avoid the trivial solution. However, the possible $\mathbf{\Omega}$ to be learned is restricted due to a rather arbitrary constraint enforced for the learning problem. In [7], the Analysis K-SVD algorithm is proposed for analysis dictionary learning. A data set, i.e., the sub-matrix of $\mathbf{Y}$, can be obtained, with its columns corresponding to that of $\hat{\mathbf{X}}$, where $\mathbf{Y} = [\mathbf{y}_1\ \mathbf{y}_2\ ...\ \mathbf{y}_K] \in R^{M\times K}$ is the observed signals 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. 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 $\mathbf{Y}$. The smallest singular values of this sub-matrix are then used to update $\mathbf{\Omega}$. The limitation of this algorithm is that it needs to pre-estimate the signal $\mathbf{X}$ in order to learn the dictionary. In [8], a sequential minimal eigenvalue based ADL algorithm was proposed by considering the orthogonality between the

978-1-4799-1180-6/13/$31.00

row of the analysis dictionary $\mathbf{\Omega}$ and a sub-set of training signals $\mathbf{X}$. 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. The computational cost of this method, however, increases with the increase of the number of rows in $\mathbf{\Omega}$.

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, estimation of $\mathbf{X}$ is computationally slow and also becomes unreliable with the increase of noise in $\mathbf{Y}$. In [9], a subset pursuit algorithm has been proposed for analysis dictionary learning (SP-ADL) which directly employs the observed data to compute the approximate analysis sparse representation of the original signals, without having to pre-estimate $\mathbf{X}$. The analysis sparse representation can be exploited to assign the observed data into multiple subsets, which are then used for updating the analysis dictionary. More specifically, the algorithm exploits the analysis representation of $\mathbf{Y}$ to obtain the subset $\mathbf{Y}_j$, rather than using $\hat{\mathbf{X}}_j$ to determine $\mathbf{Y}_j$, where $\mathbf{Y}_j$ is the subset of $\mathbf{Y}$ that corresponds to the *j*-th row of $\mathbf{\Omega}$. In this paper, we focus on the analysis model and use the K-plane clustering algorithm to learn the analysis operator $\mathbf{\Omega}$ directly from the observed signals $\mathbf{Y}$. As a result, it is faster than the Analysis K-SVD. Different from the SP-ADL algorithm mentioned above, the proposed algorithm is an adaptive clustering algorithm, offering a new way for learning the analysis dictionaries.

The paper is organized as follows. In section 2, the sparse analysis model is introduced. In section 3, the K-plane clustering algorithm is described. In section 4, the experiments on natural image denoising is given, followed by the conclusions in section 5.

2. THE ANALYSIS SPARSE MODEL

As described above, the analysis model uses the possibly redundant analysis operator $\mathbf{\Omega}$. This model assumes that the vector $\mathbf{\Omega x} = \mathbf{z}$ should be sparse [9]. In the analysis model the signal $\mathbf{x}$ is characterized by the zeros of the vector $\mathbf{\Omega x}$, and these zeros define the subspace that the signal belongs to. The co-sparsity l of the co-sparse analysis model is

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

The co-sparsity l denotes the zeros of the vector $\mathbf{\Omega x}$, implying that l rows in $\mathbf{\Omega}$ are orthogonal to the signal $\mathbf{x}$. One can observe that the larger the l, the more co-sparse the $\mathbf{x}$.

According to (1), the task of ADL can be formed as

$$\min \|\mathbf{\Omega x}\|_0 \tag{2}$$

This problem is NP-hard [10]. Just like in the synthesis case, one might replace the ℓ_0 quasi-norm with the ℓ_1 norm

$$\min \|\mathbf{\Omega x}\|_1 \tag{3}$$

It sums the absolute values of a vector. The analysis model implies that l rows in $\mathbf{\Omega}$ are orthogonal to the signal $\mathbf{x}$, and these rows define the co-support Λ, i.e. $\mathbf{\Omega}_\Lambda \mathbf{x} \approx 0$, where $\mathbf{\Omega}_\Lambda$ is a sub-matrix of $\mathbf{\Omega}$ that contains the rows from $\mathbf{\Omega}$ indexed by Λ.

In the analysis model, if the true co-support Λ is known, the signal $\mathbf{x}$ can be recovered from its noisy version $\mathbf{y} = \mathbf{x} + \mathbf{v}$

$$\hat{\mathbf{x}} = \left(\mathbf{I} - \mathbf{\Omega}_\Lambda^\dagger \mathbf{\Omega}_\Lambda\right)\mathbf{y} \tag{4}$$

3. THE K-PLANE CLUSTERING ALGORITHM FOR ANALYSIS DICTIONARY LEARNING

In this section, we present a K-plane clustering algorithm for analysis dictionary learning directly using the observed data without having to pre-estimate $\mathbf{X}$, where $\mathbf{X} = [\mathbf{x}_1\ \mathbf{x}_2\ ...\ \mathbf{x}_K] \in R^{M\times K}$ contains the original signals. The model that we consider in our work can be written as

$$\mathbf{z}_i = \mathbf{\Omega y}_i \tag{5}$$

where $\mathbf{z}_i = [z_{1i}\ \ z_{2i}\ \ \cdots\ \ z_{Pi}]^T \in R^{P\times 1}$ is the analysis representation of $\mathbf{y}_i$.

Based on the fact that the quantity l denotes the number of zeros in the vector $\mathbf{\Omega y}_i$, we can assume that l rows in $\mathbf{\Omega}$ are approximately orthogonal to the signal $\mathbf{y}_i$. These rows define the co-support Λ, i.e. $\mathbf{\Omega}_\Lambda \mathbf{y}_i = 0$. The optimisation problem that we aim to solve is therefore written as

$$\hat{\mathbf{\Omega}}_\Lambda = \arg\min_{\mathbf{\Omega}_\Lambda} \|\mathbf{\Omega}_\Lambda \mathbf{y}_i\|_1^2 \tag{6}$$

In the analysis sparse model, the observed data have the characteristic that they are co-planer, i.e. many columns of the observed data form multi-dimensional subspaces (hyperplanes). To reduce the effect of outliers, we normalize all observation vectors $\mathbf{y}_i$ in preprocessing stage to a unit length. The samples in a cluster are distributed on an ortho-drome. The normal directions of the K-dimensional concentration hyper-planes can be estimated using the K-plane clustering algorithm, and then the rows of the analysis dictionary, which are the normal vectors of the hyper-planes, can be obtained.

Let $\mathbf{y}_i$ be the *i*th normalized column of $\mathbf{Y}$. To fit a hyperplane to the samples $\mathbf{y}_i$, the normal directions of the hyperplane is modified as follows:

$$\mathbf{\Omega}_\Lambda^{(t+1)} = \mathbf{\Omega}_\Lambda^t - \eta(t)\langle \mathbf{\Omega}_\Lambda^t, \mathbf{y}_i\rangle \mathbf{y}_i \tag{7}$$

where $0 < \eta \leq 1$ is a learning rate which controls convergence properties of the learning process and $\langle\cdot\rangle$ is an inner product operation. If $\eta \doteq 0$, $\mathbf{\Omega}_\Lambda$ is not modified. If $\eta \doteq 1$, $\mathbf{\Omega}_\Lambda$ is projected to the orthogonal complement of the space spanned by $\mathbf{y}_i$. Then, $\mathbf{\Omega}_\Lambda$ and $\mathbf{y}_i$ are perpendicular. In other words, $\mathbf{y}_i$ lies on the *i*th hyperplane. If $0 < \eta < 1$, $\mathbf{\Omega}_\Lambda$ is modified to be near orthogonal to $\mathbf{y}_i$. In practice, the learning

rate decreases monotonically with respect to learning iterations *t*. Therefor, $\eta(t)$ is modified as follows:

$$\eta(t+1) = \max(\eta(t) - \eta_{dec}, \eta_{min}) \tag{8}$$

where η_{dec} is the adjustment, and η_{min} is the minimum of $\eta(t)$. In our experiments, we choose empirically $\eta_{dec} = 0.00025$, $\eta_{min} = 0.001$, and the initial value of $\eta(t) = 0.5$.

In the K-plane clustering algorithm, we consider $\mathbf{\Omega}_\Lambda$ as a block of $\mathbf{\Omega}$ so that the ADL is computationally efficient. We can remove columns $\mathbf{y}_i$ from the matrix $\mathbf{Y}$, if the columns satisfy the condition:

$$\mathbf{\Omega}_\Lambda \mathbf{y}_i = 0 \tag{9}$$

The proposed algorithm is summarised in Algorithm 1.

4. EXPERIMENTS ON NATURAL IMAGE DENOISING

To validate the proposed algorithm, we present results of the experiments. In the experiments we consider the natural images denoising problem. $\mathbf{\Omega}_0 \in R^{P\times M}$ is randomly generated and each row of $\mathbf{\Omega}_0$ is normalized. $\mathbf{Y} \in R^{M\times K}$ is the noisy version of the signals $\mathbf{X}$, and each column of $\mathbf{Y}$ is normalized.

Algorithm 1: K-plane clustering for ADL

Input: Observed data $\mathbf{Y} \in R^{M\times K}$, the initial dictionary $\mathbf{\Omega}_0 \in R^{P\times M}$, the co-sparsity l, 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 $\mathbf{Y}' = [\mathbf{y}'_1 ... \mathbf{y}'_K] \in R^{M\times K}$, and set $\eta(1) = 0.5$
For $t = 1...T$ **do**
  For $i = 1...K$ **do**

- Compute $\mathbf{z}_i = \mathbf{\Omega}\mathbf{y}'_i$, and select l numbers of $|\mathbf{z}_i|$ which have the smallest values and obtain the co-support Λ_i. Choose the rows from the co-support Λ_i, i.e. $\mathbf{\Omega}_{\Lambda_i}\mathbf{y}'_i = 0$, where $\mathbf{\Omega}_{\Lambda_i}$ is a sub-matrix of $\mathbf{\Omega}$ that contains the rows from $\mathbf{\Omega}$ indexed by Λ_i.

- Estimate the analysis dictionary using the following formula.

$$\mathbf{\Omega}_\Lambda^{(t+1)} = \mathbf{\Omega}_\Lambda^t - \eta(t)\langle \mathbf{\Omega}_\Lambda^t, \mathbf{y}'_i \rangle \mathbf{y}'_i$$

  End

Update $\eta(t+1) = \max(\eta(t) - \eta_{dec}, \eta_{min})$.
Remove columns $\mathbf{y}'_i$ from the matrix $\mathbf{Y}$, if the columns conform the condition: $\mathbf{\Omega}_\Lambda \mathbf{y}'_i = 0$.
End

In this experiment, using the test set consisting of three images commonly used in denoising (Lena, house and peppers) [7, 11], we perform denoising experiments and compare the performance of the K-plane clustering algorithm with that of the Analysis K-SVD [7] and the SP-ADL [9] algorithms on the same experimental protocol. The denoising performance is evaluated by the peak signal to noise ratio (PSNR) defined as

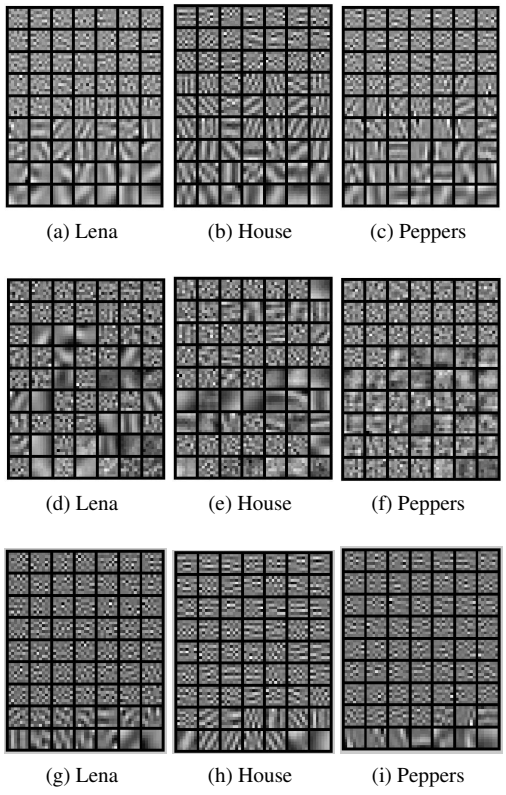
$$\text{PSNR} = 10\log_{10}\left(\frac{255^2 \times M \times K}{\Sigma_{i=1}^{M}\Sigma_{j=1}^{K}(\mathbf{y}_{ij} - \mathbf{x}_{ij})^2}\right). \tag{10}$$

where $M = 49$ and $K = 20000$. The dictionary of size 63×49 is created by using a training set of size 20,000 of 7×7 image patches. Noise of different levels σ, varying from 5 to 20, is added to these image patches. We also assume that the co-sparsity $l = M - 7$. According to the experiments, we choose $\eta_{dec} = 0.00025$, $\eta_{min} = 0.001$, and the initial value of $\eta(t) = 0.5$. In this experiment, if $\mathbf{\Omega}_\Lambda \mathbf{y}_i < 10^{-8}$, we can remove columns $\mathbf{y}_i$ from the matrix $\mathbf{Y}$. We apply 50 iterations to learn the analysis dictionary for these algorithms. The examples of the learned dictionaries are shown in Figure 1 for $\sigma = 10$.

The learned dictionaries are employed for patch-based image denoising, and the results are presented in Table 1, from which we can observe that the performance of the K-plane clustering algorithm is almost the same as that of the Analysis K-SVD and a little lower than that of the SP-ADL algorithm. However, the K-plane clustering algorithm is faster than the Analysis K-SVD, and slower than SP-ADL algorithm. In the same simulation environment, when $\sigma = 20$, the K-plane clustering algorithm took about 4237 (Lena), 2500 (House) and 3620 (Peppers) seconds to learn the dictionary, respectively, while the Analysis K-SVD took about 8187 (Lena), 7613 (House) and 8023 (Peppers) seconds, respectivly, and the SP-ADL algorithm took about 575 (Lena), 587 (House) and 577 (Peppers) seconds, respectivly (Computer OS: Windows 7, CPU: Intel Core i5-2410M @ 2.30GHz, RAM: 2G).

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
		K-plane clustering	38.20	38.68	37.23
		SP-ADL	38.40	38.90	37.73
10	28.13	Analysis K-SVD	34.85	35.27	33.80
		K-plane clustering	34.82	35.06	33.34
		SP-ADL	35.13	35.23	33.83
15	24.61	Analysis K-SVD	32.59	33.00	31.31
		K-plane clustering	33.02	33.00	31.14
		SP-ADL	33.22	33.25	31.61
20	22.11	Analysis K-SVD	31.42	31.49	29.80
		K-plane clustering	31.52	31.57	29.47
		SP-ADL	31.90	31.88	30.05



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

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

(g) Lena (h) House (i) Peppers

Fig. 1: The learned dictionaries of size 63×49 by using the Analysis K-SVD, the K-plane clustering algorithm, and the SP-ADL algorithm on the three images with noise level $\sigma = 10$. The results of the Analysis K-SVD are shown in (a) Lena, (b) House and (c) Peppers, and the results of the K-plane clustering algorithm are shown in (d) Lena, (e) House and (f) Peppers, with the others showing the results of the SP-ADL algorithm.

5. CONCLUSION

In this paper, a novel algorithm based on K-plane clustering has been proposed for analysis dictionary learning. In the K-plane clustering algorithm, we directly use the observed data $\mathbf{Y}$ to estimate the rows of the analysis dictionary which are the normal vectors of the hyperplanes. As a result, it is faster than the Analysis K-SVD, and for different noise levels also provides competitive denoising performance, however, it is slower than the SP-ADL algorithm. The experiments have shown that K-plane clustering algorithm is well suited for the analysis dictionary learning.

6. ACKNOWLEDGEMENTS

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

7. REFERENCES

[1] Y. Washizawa. and A. Cichocki., "Sparse blind identification and separation by using adaptive k-orthodrome clustering," *Neurocomputing*, pp. 2321 –2329, March 2008.

[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.

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

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

[6] M. Yaghoobi, S. Nam, R. Gribonval, and M. E. Davies, "Analysis operator learning for overcomplete cosparse representations," in *Proc. European Signal Processing Conference*, Barcelona, Spain, 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 *Proc. European Signal Processing Conference*, 2011, pp. 1465–1469.

[9] Y. Zhang, H. Wang, T. Yu, and W. Wang, "Subset pursuit for analysis dictionary learning," in *Proc. European Signal Processing Conference*, 2013.

[10] 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, 2006.

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