RELATIVE ATTRIBUTE GUIDED DICTIONARY LEARNING

Mohammadreza Babaee, Thomas Wolf, Gerhard Rigoll

Institute for Human-Machine Communication, TU München
Theresienstrae 90, 80333 München, Germany

ABSTRACT

A discriminative dictionary learning algorithm is proposed to find sparse signal representations using relative attributes as the available semantic information. In contrast, existing (discriminative) dictionary learning approaches utilize binary label information to enhance the discriminative property of the signal reconstruction residual, the sparse coding vectors or both. Compared to binary attributes or labels, relative attributes contain richer semantic information where the data is annotated with the attributes' strength. In this paper we use the relative attributes of training data indirectly to learn a discriminative dictionary. Precisely, we incorporate a rank function for the attributes in the dictionary learning process. In order to assess the quality of the obtained signals, we apply k-means clustering algorithm to the obtained signals and measure the clustering performance. Experimental results conducted on three datasets confirm that the proposed approach outperforms the state-of-the-art label based dictionary learning algorithms.

***Index Terms*—** Relative Attributes, Dictionary Learning, Clustering

1. INTRODUCTION

The concept of sparse coding has become very popular in many fields of engineering such as signal analysis and processing [1], clustering and classification [2, 3, 4], and face recognition [5]. The idea behind a sparse representation is to approximate a signal by a linear combination of a small set of elements from a so called over-complete dictionary. The coding vector specifying the linear combination is the sparse representation of the original input signal.

Considering a set of n input signals $\mathbf{Y} \in \mathbb{R}^{p\times n}$, the goal is to find a dictionary $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_n] \in \mathbb{R}^{p\times k}$ and the sparse representation $\mathbf{X} \in \mathbb{R}^{k\times n}$ such that $\mathbf{Y} \cong \mathbf{DX}$, where the term *over-complete* indicates $k > n$. Dictionaries can either be predefined as in the form of wavelets [6], or be learned from observations [7, 8, 9]. Also, many approaches have been developed to impose discriminative capabilities onto the dictionary learning process. However, those methods use binary label information to acquire discriminative behavior.

In this work, we present an approach that utilizes relative attributes instead of binary labels to enhance the discriminative property of the dictionary. Relative attributes, as described in [10], represent the strength of a set of predefined attributes rather than only their appearance. This way of describing often seems more natural to humans. For example, is a gazelle a big animal? That is hard to say. In the context of relative attributes one can say a gazelle is bigger than a cat but smaller than an elephant. Just as previous discriminative dictionary learning approaches that use binary label information to enhance their discriminative capabilities, we incorporate relative attributes into the dictionary learning process to give the learning process some semantic information.

The rest of paper is organized as follows. In Section 2, related work in the field of dictionary learning and relative attributes is presented. Section 3 illustrates the proposed algorithm and in Section 4 the experiments and results are presented. Finally, we provide a summary and draw our conclusion in Section 5.

2. RELATED WORK

The first approaches in the field of (reconstructive) dictionary learning are the K-SVD algorithm [7] and the Method of Optimal Direction (MOD) [11], where no semantic information is used in the learning process. An additional example for the usage of sparse representation is the sparse representation based classification (SRC) [5] in which the dictionary is built directly from the training data. A large field in dictionary learning area is called *Discriminative Dictionary Learning* (DDL), where either the discriminative property of the signal reconstruction residual, or the discriminative property of the sparse representation itself is enhanced. Approaches with a focus on the reconstruction residual are the work of Ramirez *et al.* [12] which includes a structured incoherence term to find independent sub-directories for each class and the work of Gao *et al.* [13] where sub-dictionaries for the different classes are learned as well as a shared dictionary over all classes.

Methods aiming at finding discriminative coding vectors learn the dictionary and a classifier simultaneously. In the work of Zhang *et al.* [8], the K-SVD algorithm is extended by a linear classifier. Jiang *et al.* [14] included an additional discriminative regularizer to come up with the so called label consistent KSVD (LC-KSVD) algorithm. Both of these algorithms show good results for classification and face recognition tasks. The approach of Yang *et al.* [15] combines the two types of DDL by taking the discriminative capabilities of the reconstruction residual and the sparse representation into account. Therefore, class specific sub-dictionaries are learned while maintaining discriminative coding vectors by applying the Fisher discrimination criterion. In the recent work of Cai *et al.* [9], a new so called Support Vector Guided Dictionary Learning (SVGDL) algorithm is presented where the discrimination term consists of a weighted summation over squared distances between the pairs of coding vectors. The algorithm automatically assigns non-zero weights to critical vector pairs (the support vectors) leading to a generalized good performance in pattern recognition tasks.

2.1. Background

For the general problem formulation we assume $\boldsymbol{Y} = [\boldsymbol{y_1}, \boldsymbol{y_2}, ...\boldsymbol{y_n}]$ to be the set of p-dimensional input signals, each belonging to one of C (hidden) classes, $\boldsymbol{X} = [\boldsymbol{x_1}, \boldsymbol{x_2}, ..., \boldsymbol{x_n}]$ to be their corresponding k-dimensional sparse representation and $\boldsymbol{D} \in \mathbb{R}^{p\times k}$ to be the dictionary. Therefore, we formulate the dictionary learning problem as

$$< \boldsymbol{D}, \boldsymbol{X} >= \underset{\boldsymbol{D},\boldsymbol{X}}{\arg\min} \|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_2^2 + \lambda_1 \|\boldsymbol{X}\|_1, \quad (1)$$

with the regularization parameter λ_1. In order to take the relative attributes into account the objective function can be extended with an additional term $\mathcal{L}(\boldsymbol{X})$, where

$$< \boldsymbol{D}, \boldsymbol{X} >= \underset{\boldsymbol{D},\boldsymbol{X}}{\arg\min} \|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_2^2 + \lambda_1 \|\boldsymbol{X}\|_1 + \lambda_2 \mathcal{L}(\boldsymbol{X}). \quad (2)$$

As additional information, the strength of M predefined attributes, the so called relative attributes [10], for the input signals are available. Those attributes, in contrast to binary labels, represent the strength of a property instead of its presence. The idea in learning relative attributes, assuming there are M attributes $\mathcal{A} = \{a_m\}$, is to learn M ranking functions $\boldsymbol{w_m}$ for $m = 1..M$. Therefore, the predicted relative attributes are computed by

$$r_m(\boldsymbol{x_i}) = \boldsymbol{w_m}^\intercal \boldsymbol{x_i}, \quad (3)$$

such that the maximum number of the following constraints is satisfied:

$$\forall (i,j) \in \mathcal{O}_m : \boldsymbol{w_m}^\intercal \boldsymbol{x_i} > \boldsymbol{w_m}^\intercal \boldsymbol{x_j}, \quad (4)$$

$$\forall (i,j) \in \mathcal{S}_m : \boldsymbol{w_m}^\intercal \boldsymbol{x_i} \approx \boldsymbol{w_m}^\intercal \boldsymbol{x_j}, \quad (5)$$

whereby $\mathcal{O}_m = \{(i,j)\}$ is a set of ordered signal pairs with signal i having a stronger presence of attribute a_m than signal j and $\mathcal{S}_m = \{(i,j)\}$ being a set of unordered pairs where signal i and j have about the same presence of attribute a_m. The work of Parikh *et al.* [10] provides us with the convenient *RankSVM* function that returns the ranking vector $\boldsymbol{w_m}$ for a set of input images and their relative ordering. This information can further be used in the objective function in Eq. (2).

3. RELATIVE ATTRIBUTE GUIDED DICTIONARY LEARNING

The *RankSVM* function maps the original input signal ($\boldsymbol{y_i}$) to a point (q_i) in a so-called relative attribute space. Additionally, we assume that there exists a linear transformation (i.e., $\boldsymbol{A}$) that maps the sparse signal ($\boldsymbol{x_i}$) to the point q_i (see Figure 1 and Eq. (6)). First, we define the matrix $\boldsymbol{Q} \in \mathbb{R}^{n\times M}$

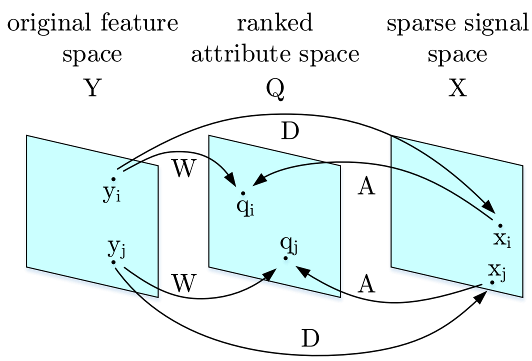

Fig. 1. Illustration of signal transformations. The goal is to transform $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$ as close as possible to q_i and q_j.

with the elements $q_{im} = r_m(\boldsymbol{y_i})$ that contains the strength of the (relative) attributes of all signals in $\boldsymbol{Y}$. In order to find the transformation of $\boldsymbol{Y}$ into $\boldsymbol{Q}$ we apply the *RankSVM* function known from [10] onto the original input signal and obtain the weighting matrix $W = [\boldsymbol{w_1}^\intercal; \boldsymbol{w_2}^\intercal; ...; \boldsymbol{w_M}^\intercal]$. The objective is to find a matrix $\boldsymbol{A}$, which transforms the sparse representation of the signals into their corresponding relative attribute representations $\boldsymbol{Q}$ with a minimum distance between $\boldsymbol{w_m}^\intercal \boldsymbol{y_i}$ and $\boldsymbol{a_m}^\intercal \boldsymbol{x_i}$.

$$\underset{\boldsymbol{A}}{\arg\min} \|\boldsymbol{Q} - \boldsymbol{A}\boldsymbol{X}\|_2^2 = \underset{\boldsymbol{A}}{\arg\min} \|\boldsymbol{W}\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}\|_2^2. \quad (6)$$

By using Eq. (6) in Eq. (2) as a loss term we get the formulation

$$< \boldsymbol{D}, \boldsymbol{X} >= \underset{\boldsymbol{D},\boldsymbol{X},\boldsymbol{A}}{\arg\min} \|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_2^2 + \lambda_1 \|\boldsymbol{X}\|_1 + \lambda_2 \|\boldsymbol{W}\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}\|_2^2. \quad (7)$$

From the first part of the equation we can see that $\boldsymbol{Y} \cong \boldsymbol{D}\boldsymbol{X}$. If $\boldsymbol{Y}$ in the loss term for the relative attributes is approximated by $\boldsymbol{D}\boldsymbol{X}$. Then the equation becomes

$$< \boldsymbol{D}, \boldsymbol{X} >= \underset{\boldsymbol{D},\boldsymbol{X},\boldsymbol{A}}{\arg\min} \|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_2^2 + \lambda_1 \|\boldsymbol{X}\|_1 + \lambda_2 \|\boldsymbol{W}\boldsymbol{D}\boldsymbol{X} - \boldsymbol{A}\boldsymbol{X}\|_2^2. \quad (8)$$

The third term of Eq. (8) is minimized if $\boldsymbol{A} = \boldsymbol{W}\boldsymbol{D}$. This information can be used to eliminate $\boldsymbol{A}$ from Eq. (7) to arrive at the final objective function

$$
\begin{aligned}
< \boldsymbol{D}, \boldsymbol{X} >= \underset{\boldsymbol{D},\boldsymbol{X}}{\arg\min} \, &\|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_2^2 + \lambda_1 \|\boldsymbol{X}\|_1 \\
&+ \lambda_2 \|\boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X})\|_2^2 .
\end{aligned} \tag{9}
$$

In order to find a closed form solution for (9 and to reduce computational complexity the term $\|\boldsymbol{X}\|_1$ is replaced with $\|\boldsymbol{X}\|_2^2$. This can be justified (as in [9]) because the goal is now to learn a discriminative dictionary and not to obtain sparse signals . However, once the dictionary is learned, a sparse representation can obtained by the orthogonal matching pursuit [16]. The final objective function is

$$
\begin{aligned}
< \boldsymbol{D}, \boldsymbol{X} >= \underset{\boldsymbol{D},\boldsymbol{X}}{\arg\min} \, &\|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_2^2 + \lambda_1 \|\boldsymbol{X}\|_2^2 \\
&+ \lambda_2 \|\boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X})\|_2^2 .
\end{aligned} \tag{10}
$$

This equation is not a jointly convex optimization problem, so $\boldsymbol{X}$ and $\boldsymbol{D}$ are optimized alternately. The update rules for $\boldsymbol{D}$ and $\boldsymbol{X}$ are found by deriving the objective function in (11) and setting the derivatives to zero. Precisely,

$$
O = \|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_2^2 + \lambda_1 \|\boldsymbol{X}\|_2^2 + \lambda_2 \|\boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X})\|_2^2 \tag{11}
$$

where

$$
\begin{aligned}
\frac{\partial O}{\partial \boldsymbol{D}} =& 2(\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X})\boldsymbol{X}^\intercal \\
&+ 2\lambda_2 \boldsymbol{W}^\intercal(\boldsymbol{W}\boldsymbol{Y} - \boldsymbol{W}\boldsymbol{D}\boldsymbol{X})\boldsymbol{X}^\intercal = 0 \\
\Rightarrow \boldsymbol{D} =& \boldsymbol{Y}(\boldsymbol{X}^\intercal\boldsymbol{X})^{-1}\boldsymbol{X}^\intercal
\end{aligned} \tag{12}
$$

and

$$
\begin{aligned}
\frac{\partial O}{\partial \boldsymbol{X}} =& -2\boldsymbol{D}^\intercal(\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}) + 2\lambda_1\boldsymbol{X} \\
&- 2\lambda_2 \boldsymbol{D}^\intercal\boldsymbol{W}^\intercal(\boldsymbol{W}\boldsymbol{Y} - \boldsymbol{W}\boldsymbol{D}\boldsymbol{X}) = 0 \\
\Rightarrow \boldsymbol{X} =& (\boldsymbol{D}^\intercal\boldsymbol{D} + \lambda_1\boldsymbol{I} + \lambda_2\boldsymbol{D}^\intercal\boldsymbol{W}^\intercal\boldsymbol{W}\boldsymbol{D})^{-1} \\
&* (\boldsymbol{D}^\intercal\boldsymbol{Y} + \lambda_2\boldsymbol{D}^\intercal\boldsymbol{W}^\intercal\boldsymbol{Y}).
\end{aligned} \tag{13}
$$

The complete algorithm works as follows. Initially the *RankSVM* [10] function is used to learn the ranking matrix $\boldsymbol{W}$ from the original input data $\boldsymbol{Y}$ and their relative ordering (i.e., sets $\mathcal{O}_m, \mathcal{S}_m$). The initial dictionary $\boldsymbol{D}$ and the sparse representation of the data is obtained by first building a dictionary from **randomly** chosen input signals and then applying the KSVD-algorithm [7]. Afterwards, the dictionary and the sparse representation are optimized alternately until convergence. We first optimize $\boldsymbol{D}$ with respect to the initial representation $\boldsymbol{X}$. Then $\boldsymbol{X}$ is updated depending on the new Dictionary $\boldsymbol{D}$, and so forth. In order to avoid scaling issues that may affect the convergence, the dictionary is L_2 normalized column-wise. The structure of the algorithm can be seen in Algorithm 1.

Algorithm 1 Relative Attribute Guided Dictionary Learning

Require: Original signal $\boldsymbol{Y}$, sets of ordered ($\mathcal{O}_m$) and unorderd images ($\mathcal{S}_m$)
Ensure: Dictionary $\boldsymbol{D}$

```
W ← RankSVM(Y, O_m, S_m)
D_init ← rndperm(Y)
D, X ← KSVD(D_init, Y)
for i = 0 to convergence do
    D ← Y(XᵀX)⁻¹Xᵀ
    D ← normcol(D)
    X ← (DᵀD + λ₁I + λ₂DᵀWᵀWD)⁻¹(DᵀY − λ₂DᵀWᵀY)
end for
```

4. EXPERIMENTS

4.1. Datasets

In order to assess the quality of the learned dictionary obtained from the proposed algorithm, we purpose a clustering task for three public available datasets, namely Public Figure Face (PubFig) [17], Outdoor Scene Recognition (OSR) [18] and Shoes [19]. These sets have been chosen, since they are the only ones known to us with annotated relative attributes. Some sample images of each dataset are presented in Fig. 2.

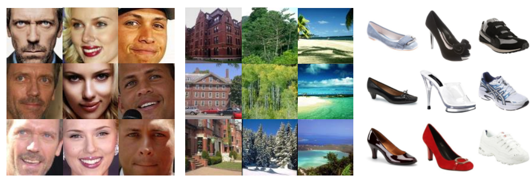

Fig. 2. Example images from the PubFig (left), OSR (middle) and Shoes (right) datasets.

a) The **PubFig** dataset contains 772 images from 8 different identities defined by the 512 dimensional GIST [18] features and is split into 241 training images and 531 test images.

b) The **OSR** set consists of 2688 images from 8 categories described again by the 512 dimensional GIST [18] features split into 240 training and 2488 testing images.

c) In the **Shoes** dataset there are 14658 images from 10 different types. Out of this set 240 images were used for training and 1579 for testing. The images are described by 960 dimensional GIST [18] features.

Additionally, tests were conducted to find the optimal values for λ_1 and λ_2. Therefore, different fixed values were chosen for λ_1 while iterating over candidates for λ_2. The chosen values are $\lambda_1 = 0.01$ and $\lambda_2 = 1$ for the Pubfig dataset, $\lambda_1 = 0.1$ and $\lambda_2 = 0.01$ for the OSR dataset and $\lambda_2 = 0.001$ and $\lambda_2 = 0.1$ for the Shoes dataset.

4.2. Evaluation Metrics

In order to quantify the clustering capabilities of the sparse representation, the k-means algorithm [20] is applied and the accuracy (AC) and the normalized Mutual Information (nMI) metrics [21] are computed. Furthermore, the sparse representation is obtained from the learned dictionary by approximating $\boldsymbol{X}$ in the error-constrained sparse coding problem, given by Eq. (14), with the help of the OMP-Box Matlab toolbox [16], where the reconstruction error from the training phase is chosen as ε. Those signals can then be used for the clustering.

$$\hat{\boldsymbol{X}} = \arg\min_{\boldsymbol{X}} \|\boldsymbol{X}\|_0 \quad \text{s.t.} \quad \boldsymbol{X} = \|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_2^2 \leq \varepsilon \tag{14}$$

4.3. Results

As a benchmark for the results, different supervised and unsupervised (discriminative) dictionary learning techniques are used, namely (1) KSVD [7], (2) SRC [5] as unsupervised techniques and (3) LC-KSVD [8], (4) FDDL [15], (5) SVGDL [9] as supervised techniques. The results were compared by their performance for full label information for varying dictionary sizes. Fig. 3 shows the behavior of the algorithms for an increasing the dictionary size with the complete training data available. The dictionary sizes used were [16, 40, 80, 120, 160, 240] for the PubFig and OSR dataset and [20, 50, 100, 140] for the Shoes dataset, which corresponds to [2, 5, 10, 15, 20, 30] and [2, 5, 10, 14] atoms per class. The number of atoms per class are constrained by the partition of the data into training and testing (for the Shoes dataset one class only includes 14 training samples). One should notice that the FDDL algorithm cannot use all training data, since the dictionary size restricts the size of the training samples. Therefore, only in the last test case the algorithm uses the complete training information.
The results show that for the proposed algorithm the accuracy increases with the dictionary size, up to values exceeding the compared algorithms. However, for the OSR and Shoes dataset and an increasing dictionary size the SVGDL and FDDL produce comparable results. Additionally, the runtime of the training phase was analyzed, where the proposed algorithm outperformed all contestants with an average training time of 1.75 s.

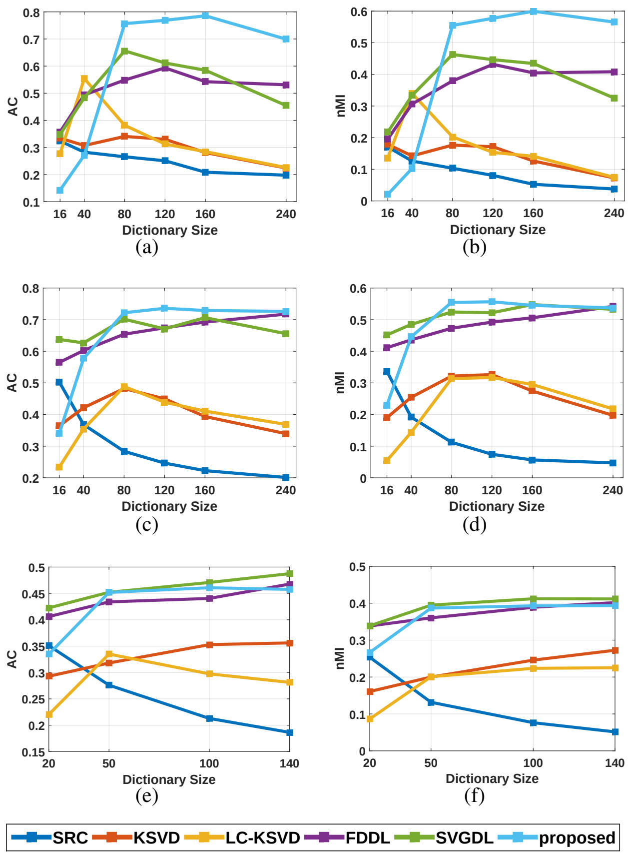

Fig. 3. Clustering results for all three datasets for increasing dictionary sizes. The first and second column represent the Accuracy (AC) and normalized MI (nMI), respectively. The first, second, and third rows are the results of PubFig, OSR, and Shoes datasets, respectively.

5. CONCLUSION

We have presented a novel discriminative dictionary learning algorithm that utilizes relative attributes instead of binary labels. The relative attributes provide a much richer semantic information to improve the discriminative property of the dictionary and eventually the sparse representation of the input signal. Instead of using relative attributes of the images directly, we use the learned ranking functions in the learning process. The ranking functions transform the original features into a relative attribute space and therefore, we aim to transform the sparse signal linearly into this attribute space. This can be achieved by adding an additional loss term to the objective function of a standard dictionary learning problem. The results not only show promising results that are comparable to modern DDL approaches, but also has low computational time for learning the dictionary outperforming all of the compared approaches.

6. REFERENCES

[1] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*. ACM, 2009, pp. 689–696.

[2] M. Elad, *Sparse and redundant representations: from theory to applications in signal and image processing*, Springer Science & Business Media, 2010.

[3] M. Babaee, S. Tsoukalas, M. Babaee, G. Rigoll, and M. Datcu, "Discriminative nonnegative matrix factorization for dimensionality reduction," *Neurocomputing*, vol. 173, pp. 212–223, 2016.

[4] M. Babaee, S. Tsoukalas, G. Rigoll, and M. Datcu, "Immersive visualization of visual data using nonnegative matrix factorization," *Neurocomputing*, vol. 173, pp. 245–255, 2016.

[5] J. Wright, A. Y Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 2, pp. 210–227, 2009.

[6] S. Mallat, *A wavelet tour of signal processing*, Academic press, 1999.

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

[8] Q. Zhang and B. Li, "Discriminative k-svd for dictionary learning in face recognition," in *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*. IEEE, 2010, pp. 2691–2698.

[9] S. Cai, W. Zuo, L. Zhang, X. Feng, and P. Wang, "Support vector guided dictionary learning," in *Computer Vision–ECCV 2014*, pp. 624–639. Springer, 2014.

[10] D. Parikh and K. Grauman, "Relative attributes," in *Computer Vision (ICCV), IEEE International Conference on*. IEEE, 2011, pp. 503–510.

[11] K. Engan, S. O. Aase, and J. Husoy, "Frame based signal compression using method of optimal directions (mod)," in *Circuits and Systems, 1999. ISCAS'99. Proceedings of the 1999 IEEE International Symposium on*. IEEE, 1999, vol. 4, pp. 1–4.

[12] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*. IEEE, 2010, pp. 3501–3508.

[13] S. Gao, I.-H. Tsang, and Y. Ma, "Learning category-specific dictionary and shared dictionary for fine-grained image categorization," *Image Processing, IEEE Transactions on*, vol. 23, no. 2, pp. 623–634, 2014.

[14] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent k-svd," in *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*. IEEE, 2011, pp. 1697–1704.

[15] M. Yang, D. Zhang, and X. Feng, "Fisher discrimination dictionary learning for sparse representation," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 543–550.

[16] R. Rubinstein, Mi. Zibulevsky, and M. Elad, "Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit," *CS Technion*, vol. 40, no. 8, pp. 1–15, 2008.

[17] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and Simile Classifiers for Face Verification," in *IEEE International Conference on Computer Vision (ICCV)*, Oct 2009.

[18] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.

[19] A. Kovashka, D. Parikh, and K. Grauman, "Whittlesearch: Image Search with Relative Attribute Feedback," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012.

[20] J. MacQueen et al., "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Oakland, CA, USA., 1967, vol. 1, pp. 281–297.

[21] M. Babaee, R. Bahmanyar, G. Rigoll, and M. Datcu, "Farness preserving non-negative matrix factorization," in *Image Processing (ICIP), IEEE International Conference on*. IEEE, 2014, pp. 3023–3027.