https://doi.org/10.2352/ISSN.2470-1173.2017.2.VIPC-402

Graph Regularized Sparse Coding by Modified Online Dictionary Learning

[a]Lingdao Sha [1] [b]Dan Schonfeld [1] [c]Jing Wang [2]
[a] *lsha3@uic.edu,* [b] *dans@uic.edu,* [c] *jiwang12@uic.edu*
[1] *Department of Electrical and Computer Engineering, University of Illinois at Chicago*
[2] *Department of Mathematics, Statistics and Computer Science , University of Illinois at Chicago*
851 S Morgan St Chicago, Illinois 60607, USA

***Abstract*—Sparse coding - modelling data vectors as sparse linear combinations of basis elements - has been widely and successfully used in image classification, noise reduction, texture synthesis, audio processing, etc. Although traditional sparse coding with fixed dictionaries like wavelet and curvelet can produce promising results, unsupervised sparse coding has shown its advantage by optimizing the dictionary based on target data provided. However, most of the existing unsupervised sparse coding method failed to consider the high dimensional manifold information. Recently, graph regularized sparse coding has been proposed to incorporate manifold information. Better classification and clustering results have been shown compared with naive unsupervised sparse coding. The authors utilize modified feature-sign search and Lagrange dual algorithm to solve the objective function as two consecutive convex functions. This method relies on large number of iterations to get state-of-art classification and clustering results, which is computational intensive. In this paper, we proposed a novel modified online dictionary learning method which iteratively utilizes modified least angle regression and block coordinate descent method to solve the problem. Instead of getting entire coefficient matrix then generate dictionary matrix, our method updates coefficient vector and dictionary matrix in each inner iteration. Thus, efficiency and accuracy are reserved at same time.**

***Index Terms*—Image classification, image clustering, manifold learning, sparse coding, dictionary learning, online dictionary learning, least angle regression**

I. Introduction

Sparse coding enables successful representation of stimuli with only a few active coefficients. It has shown state-of-art results in ordinary signal processing tasks like image denoising [1] and restoration [2], audio [3] and video processing [4], as well as more complicated tasks like image classification [5] and image clustering [6]. When applied to natural images, sparse coding produces learned bases that can resemble the receptive fields of neurons in the visual cortex [7], which is similar to the results of Independent Component Analysis (ICA) [8] and Gabor filter [9]. Compared with other unsupervised methods like PCA and ICA, sparse coding can learn overcomplete basis sets and doesn't require statistical-independence of the dictionary prototype signals. In machine learning and statistics, slightly different matrix factorization problems such as non-negative matrix factorization, its variants [10] [11] and sparse principal component analysis [12] have been successfully used to obtain interpretable basis elements. With few basis elements needed for representation, sparse coding is a good fit for an indexing scheme that would allow quick retrieval.

Although having so many good properties, sparse coding still facing "Curse of dimensionality" when deal with high dimensional data. Tasks like image classification and image clustering can have very high dimensional feature space with each feature having a number of possible values. A reasonable thought would be sparse coding with dimensionality reduction. Manifold learning is one of the methods that deals with dimensionality reduction.

With more and more attention to sparse coding and manifold learning, Cai [13] proposed a novel graph regularized nonnegative matrix factorization method, then Gao [14] and Zheng [6] proposed graph regularized sparse coding (GraphSC), which explicitly considers the local geometrical structure of the data. Much better results have been shown by the authors compared with naive sparse coding. However, both authors use feature-sign search with Lagrange dual algorithm to solve the problem, good results rely on large number of iterations, which is computational expensive. In this paper, we proposed a novel method called modified online dictionary learning to solve the same objective function more efficiently. With modified online dictionary learning, which changed from original online dictionary learning [15] by modifying the least angle regression for coefficients optimization, we kept the feature of high efficiency from online dictionary learning as well as high accuracy from graph regularized sparse coding.

The rest of this paper is organized as follows: In Section II, we give a brief description of sparse coding problem and popular methods to solve the sparse coding problem. Section III introduces the GraphSC algorithm, as well as the novel optimization algorithm: modified online dictionary learning. Experimental results on image clustering are presented in Section IV. Finally, we conclude our paper in Section V.

Contribution

- Use K-SVD instead of PCA for preprocessing, less processing time for convergence.

- Locally linear embedding method (LLE)[16] as well as Graph Laplacian [17] for constrains.
- We came up with a novel modified online dictionary learning algorithm to solve the graph regularized sparse coding problem efficiently.

II. Sparse Coding

Given a data matrix $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_m] \in \mathbb{R}^{n\times m}$, let $\mathbf{D} = [\mathbf{d}_1, ..., \mathbf{d}_k] \in \mathbb{R}^{n\times k}$, where each $\mathbf{d}_i$ represents a basis vector in the dictionary, and $\mathbf{A} = [\alpha_1, ..., \alpha_m] \in \mathbb{R}^{k\times m}$ be the coefficient matrix, where each column is a sparse representation for a data point. A good dictionary and coefficient pair should minimize the empirical loss function, which can be represented as $\sum_{i=1}^{m} \|\mathbf{x}_i - \mathbf{D}\alpha_i\|_p$. The typical norms used for measuring the loss function are the L_p norms where $p = 1, 2$ and ∞. Here we concentrate on least square loss problems when $p = 2$.

The objective function of sparse coding can be formulated as:

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \beta \sum_{i=1}^{m} f(\alpha_i), \quad \text{s.t. } \|\mathbf{d}_i\|^2 \leqslant c, i = 1, ..., k \tag{1}$$

where f is a function to measure the sparseness of α_i and $\|\cdot\|_F$ denotes the matrix Frobenius norm.

Following [18] [19], we adopt the idea of L_1 norm instead of L_0, which can produce similar results with affordable computational cost. The objective function then becomes:

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \beta \sum_{i=1}^{m} \|\alpha_i\|_1, \quad \text{s.t. } \|\mathbf{d}_i\|^2 \leqslant c, i = 1, ..., k \tag{2}$$

Although the objective function is not convex with $\mathbf{D}$ and $\mathbf{A}$ together, it is convex with either one fixed. We iteratively optimize the objective function by minimizing over one variable with the other one fixed. Thus, it becomes an L_1-regularized least squares problem with an L_2-constrained least square problem.

III. Graph Regularized Sparse Coding (GraphSC)

A. Algorithm

To incorporate manifold information, we follow the idea of Cai [13] to build a nearest neighbor graph. Given a set of m-dimensional data points $\mathbf{x}_1, ..., \mathbf{x}_m$, we construct a nearest neighbor graph $\mathbf{G}$ with m vertices, where each vertex represents a data point. Let $\mathbf{W}$ be the weight matrix of $\mathbf{G}$. If $\mathbf{x}_i$ is among the k-nearest neighbors of $\mathbf{x}_j$ or vice versa, $W_{ij} = 1$, otherwise, $W_{ij} = 0$. We define $\mathbf{e}_i = \sum_{j=1}^{m} W_{ij}$, and $\mathbf{E} = diag(\mathbf{e}_1, ..., \mathbf{e}_m)$.

As mentioned previously, we mapping the weighted graph G to the sparse representation $\mathbf{A} = [\alpha_i, ..., \alpha_m]$, the objective function becomes:

- Laplacian Embedding fuction [17]:

$$\tfrac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} (\alpha_i - \alpha_j)^2 W_{ij} = Tr(\mathbf{A}\mathbf{L}\mathbf{A}^T) \tag{3}$$

  Where $\mathbf{L} = \mathbf{E} - \mathbf{W}$ is the Laplacian matrix.

- Locally linear Embedding (LLE) function [16]:

$$\tfrac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} |\alpha_i - \textstyle\sum_j W_{ij}\alpha_j|^2 = Tr(\mathbf{A}\mathbf{L}\mathbf{A}^T) \tag{4}$$

  Where $\mathbf{L} = (\mathbf{I} - \mathbf{W})^{\mathbf{T}}(\mathbf{I} - \mathbf{W})$, $\mathbf{I}$ is identity matrix.

Although Laplacian embedding and locally linear embedding are two different embedding methods, they share the same objective function, thus can be solved by same optimization method.

By incorporating the Laplacian or LLE regularizer into the original sparse coding, we can get the following objective function of GraphSC [6][20]:

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda Tr(\mathbf{A}\mathbf{L}\mathbf{A}^T) + \beta \sum_{i=1}^{m} \|\alpha_i\|_1 \quad \text{s.t. } \|\mathbf{d}_i\|^2 \leqslant c, \ i = 1, ..., k \tag{5}$$

where $\lambda \geqslant 0$ is the regularization parameter.

B. Coefficients Learning

In this section, we show how to solve problem (5) with fixed dictionary $\mathbf{D}$ by modified online dictionary learning algorithm.

Fixing dictionary $\mathbf{D}$, the objective function becomes:

$$\min_{\mathbf{A}} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda Tr(\mathbf{A}\mathbf{L}\mathbf{A}^T) + \beta \sum_{i=1}^{m} \|\alpha_i\|_1 \tag{6}$$

As problem (6) is convex, global minimum can be achieved[21].

With modified online dictionary learning, we update each vector $\alpha_\mathbf{i}$ individually, while keeping all the other vectors constant. In order to solve the problem by optimizing over each $\alpha_\mathbf{i}$, we rewrite problem (6) in vector form.

Reconstruction error $\|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2$ can be written as:

$$\sum_{i=1}^{m} \|\mathbf{x}_\mathbf{i} - \mathbf{D}\alpha_i\|^2 \tag{7}$$

As matrix $\mathbf{L}$ is symmetric in both Laplacian and LLE, the regularizer $Tr(\mathbf{A}\mathbf{L}\mathbf{A}^T)$ can be rewritten as:

$$Tr(\mathbf{A}\mathbf{L}\mathbf{A}^T) = Tr(\sum_{i,j=1}^{m} L_{ij}\alpha_i\alpha_j^T) = \sum_{i,j=1}^{m} L_{ij}\alpha_i^T\alpha_j \tag{8}$$

We combine reconstruction error with Laplacian or LLE regularizer, add sparsity constrain to it, the objective function becomes:

$$\min_{\alpha_i} \sum_{i=1}^{m} \|\mathbf{x}_\mathbf{i} - \mathbf{D}\alpha_i\|^2 + \lambda \sum_{i,j=1}^{m} L_{ij}\alpha_i^T\alpha_j + \beta \sum_{i=1}^{m} \|\alpha_i\|_1 \tag{9}$$

When updating α_i, the other vectors $\{\alpha_j\}_{j\neq i}$ are fixed[6] [15]. Thus, we get the following optimization problem:

$$\min_{\alpha_i} \|\mathbf{x}_\mathbf{i} - \mathbf{D}\alpha_i\|^2 + \lambda L_{ii}\alpha_i^T\alpha_j + \alpha_i^T h_i + \beta \sum_{j=1}^{k} |\alpha_i^{(j)}| \tag{10}$$

Where $h_i = 2\lambda(\sum_{j\neq i} L_{ij}\alpha_j)$ and $\alpha_i^{(j)}$ is the j-th coefficient of α_i

In Algorithm 1 of modified online dictionary learning, we keep dictionary $\mathbf{D}$ fixed, optimizing each individual coefficient α_i with all other coefficients fixed for each input data $\mathbf{x}_\mathbf{i}$. The method used is modified least angle regression which will be explained in algorithm 3.

Algorithm 1: Modified Online Dictionary Learning (MODL)

Require: $\mathbf{x} \in \mathbb{R}^m$ from $\mathbf{p}(\mathbf{x})$ ($\mathbf{x}$ sequentially aligned in $\mathbf{p}(\mathbf{x})$), $\beta \in \mathbb{R}$ (regularization parameter), $\mathbf{D}_0 \in \mathbb{R}^{m\times k}$ (initial

dictionary), T (number of samples in data set $\mathbf{p}(\mathbf{x})$).

1: $\mathbf{A}_0 \in \mathbb{R}^{k\times k} \leftarrow 0$, $\mathbf{B}_0 \in \mathbb{R}^{m\times k} \leftarrow 0$ (Reset the "past"information)

2: **for** $t = 1$ **to** T **do**

3: Draw $\mathbf{x_t}$ from $\mathbf{p}(\mathbf{x})$ (sequentially drawn)

4: Sparse coding: compute using modified LARS (Algorithm 3)

$$\alpha_t \triangleq \arg\min_{\alpha\in\mathcal{R}^k} \frac{1}{2}\|\mathbf{x}_t - \mathbf{D}_{t-1}\alpha\|_2^2 + \lambda L_{tt}\alpha^T\alpha + \alpha^T h_t + \beta\|\alpha\|_1$$

5: $\mathbf{A}_t \leftarrow \mathbf{A}_{t-1} + \alpha_t\alpha_t^T$

6: $\mathbf{B}_t \leftarrow \mathbf{B}_{t-1} + \mathbf{x}_t\alpha_t^T$

7: Compute $\mathbf{D}_t$ using Algorithm 2, with $\mathbf{D}_{t-1}$ as warm restart, so that

$$\mathbf{D}_t \triangleq \arg\min_{\mathbf{D}\in\mathcal{C}} \frac{1}{t}\sum_{i=1}^{t}(\frac{1}{2}\|\mathbf{x}_i - \mathbf{D}\alpha_i\|_2^2 + \lambda\sum_{i,j=1}^{m} L_{ij}\alpha_i^T\alpha_j + \beta\sum_{i=1}^{m}\|\alpha_i\|_1)$$

$$= \arg\min_{\mathbf{D}\in\mathcal{C}} \frac{1}{t}(\frac{1}{2}Tr(\mathbf{D}^T\mathbf{D}\mathbf{A}_t) - Tr(\mathbf{D}^T\mathbf{B}_t))$$

8:**end for**

9:Return $\mathbf{D}_T$, $\mathbf{A}$ for complete dictionary and coefficients learning

Notation: $\mathcal{C} \triangleq \{\mathbf{D} \in \mathbb{R}^{m\times k} s.t. \forall j = 1, ..., k, \mathbf{d}_j^T\mathbf{d}_j \leqslant 1\}$.

C. Dictionary Update

There are many efficient methods for updating the dictionary. Here, we use block coordinate descent [22] with warm restart, same method in [15]. One of the main advantage of this method is parameter free and no need for learning rate tuning [15].

Back to our original problem, step 7 in algorithm 1.

$$\min_{\mathbf{D}\in\mathcal{C}} \frac{1}{t}\sum_{i=1}^{t}(\frac{1}{2}\|\mathbf{x}_i - \mathbf{D}\alpha_i\|_2^2 + \lambda\sum_{i,j=1}^{m} L_{ij}\alpha_i^T\alpha_j + \beta\sum_{i=1}^{m}\|\alpha_i\|_1)$$

(11) Assume α_i doesn't depend on $\mathbf{D}$, we need to minimize:

$$\min_{D\in\mathcal{C}} \frac{1}{t}\sum_{i=1}^{t}(\frac{1}{2}\|\mathbf{x}_i - \mathbf{D}\alpha_i\|_2^2) \tag{12}$$

Setting the gradient of $\mathbf{D}_{.,j}$ to zero, we have:

$$0 = \frac{1}{t}\sum_{i=1}^{t}(\mathbf{x}_i - \mathbf{D}\alpha_i)\alpha_{j,i}$$

$$\Rightarrow = \frac{1}{t}\sum_{i=1}^{t}(\mathbf{x}_i - \mathbf{D}_{.,j}\alpha_{j,i} - \sum_{k\neq j}^{m}\mathbf{D}_{.,k}\alpha_{k,i})\alpha_{j,i}$$

$$\Rightarrow \mathbf{D}_{.,j} = \frac{1}{\underbrace{\sum_{i=1}^{t}(\alpha_{j,i})^2}_{\mathbf{A}_{j,j}}}\{\underbrace{\sum_{i=1}^{t}\mathbf{x}_i\alpha_{j,i}}_{\mathbf{B}_{.,j}} - \sum_{k\neq j}^{m}\mathbf{D}_{.,k}\underbrace{(\sum_{i=1}^{t}\alpha_{k,i}\alpha_{j,i})}_{\mathbf{A}_{k,j}}\}$$

Where $\mathbf{A} = \sum_{i=1}^{t}\alpha_i\alpha_i^T$, $\mathbf{B} = \sum_{i=1}^{t}\mathbf{x}_i\alpha_i^T$.

Algorithm 2: Dictionary Update

Require: $\mathbf{D} = [\mathbf{d_1}, ..., \mathbf{d_k}] \in \mathbb{R}^{m\times k}$ (input dictionary)
$\mathbf{A} = [\alpha_\mathbf{1}, ..., \alpha_\mathbf{k}] \in \mathbb{R}^{k\times k}$
$\mathbf{B} = [\mathbf{b_1}, ..., \mathbf{b_k}] \in \mathbb{R}^{m\times k}$

1: **Repeat**

2: **for** $j = 1$ **to** k **do**

3: update the j-th column to optimize for (10)

$$\mathbf{u}_j \leftarrow \frac{1}{\mathbf{A}[j,j]}(\mathbf{b}_j - \mathbf{D}\alpha_j) + \mathbf{d}_j$$

$$\mathbf{d}_j \leftarrow \frac{1}{max(\|\mathbf{u}_j\|_2, 1)}\mathbf{u}_j$$

4: **end for**

5:**Until Convergence**

6:Return $\mathbf{D}$, $\mathbf{A}$ (updated dictionary and coefficients)

D. Modified Least-Angle Regression

Least-Angle Regression (LARS) [23] is a regression method that provides a general version of forward selection. We follow the steps presented in [24]. In step 7 of Algorithm 3, instead of calculating the ordinary least square solution (13), we calculate the graph constrained least square solution (14) to incorporate structure information.

$$\alpha_{OLS}^{(k+1)} = (\mathbf{D}_\mathcal{A}^T\mathbf{D}_\mathcal{A})^{-1}\mathbf{D}_\mathcal{A}^T\mathbf{y} \tag{13}$$

$$\alpha_{gcOLS}^{(k+1)} = (\mathbf{D}_\mathcal{A}^T\mathbf{D}_\mathcal{A} + \lambda L_{kk}\mathbf{I})^{-1}(\mathbf{D}_\mathcal{A}^T\mathbf{x} - h_k/2) \tag{14}$$

Where $\mathbf{I}$ is identity matrix and $h_k = 2\lambda(\sum_{k\neq j} L_{kj}\alpha_j)$ from problem (10).

Algorithm 3: Modified Least-Angle Regression

1: Initialize the coefficient vector $\alpha^{(0)} = 0$ and the fitted vector $\hat{\mathbf{x}}^{(0)} = 0$.

2: Initialize the active set $\mathcal{A} = \phi$ and the inactive set $\mathcal{I} = 1, ..., p$.

3: **for** $k = 0$ **to** $p - 2$ **do**

4: Update the residual $\varepsilon = \mathbf{x} - \hat{\mathbf{x}}^{(k)}$

5: Find the maximal correlation $c = max_{i\in\mathcal{I}}|\mathbf{d}_i^T\varepsilon|$

6: Move variable corresponding to c from $\mathcal{I}$ to $\mathcal{A}$

7: Calculate the graph constrained least square solution: $\alpha_{gcOLS}^{(k+1)} = (\mathbf{D}_\mathcal{A}^T\mathbf{D}_\mathcal{A} + \lambda L_{kk}I)^{-1}(\mathbf{D}_\mathcal{A}^T\mathbf{x} - h_k/2)$ Where I is identity matrix and $h_k = 2\lambda(\sum_{k\neq j} L_{kj}\alpha_j)$

8: Calculate the current direction: $\mathtt{d} = \mathbf{D}_\mathcal{A}\alpha_{gcOLS}^{(k+1)}$ - $\hat{\mathbf{x}}^{(k)}$

9: Calculate the step length:

$$\gamma = \min_{i\in\mathcal{I}}^{+}\{\frac{\mathbf{d}_i^T\varepsilon - c}{\mathbf{d}_i^T\mathtt{d} - c}, \frac{\mathbf{d}_i^T\varepsilon + c}{\mathbf{d}_i^T\mathtt{d} + c}\},\ 0 \leq \gamma \leq 1$$

10: Update regression coefficients: $\alpha^{(k+1)} = (1-\gamma)\alpha^{(k)} + \gamma\alpha_{gcOLS}^{(k+1)}$

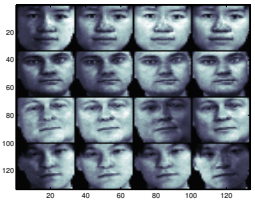
11: Update the fitted vector $\hat{\mathbf{x}}^{(k+1)} = \hat{\mathbf{x}}^{(k)} + \gamma\mathtt{d}$

12:**end for**

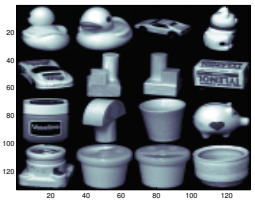
13:Let $\alpha^{(p)}$ be the full graph constrained least square solution $\alpha^{(p)} = (\mathbf{D}_\mathcal{A}^T\mathbf{D}_\mathcal{A} + \lambda L_{(p-1)(p-1)}\mathbf{I})^{-1}(\mathbf{D}_\mathcal{A}^T\mathbf{x} - h_{p-1}/2)$ where $\mathbf{I}$ is identity matrix and $h_{p-1} = 2\lambda(\sum_{p-1\neq j} L_{(p-1)j}\alpha_j)$

14: Output: the series of coefficients $\mathbf{A} = [\alpha^{(0)}, ..., \alpha^{(p)}]$

Notificatoin: $\mathbf{d}_i$ is column of Dictionary $\mathbf{D}$, $\mathtt{d}$ is direction.



(a) CMU PIE



(b) COIL

Fig. 1: Examples from 2 data sets

TABLE I: Clustering Accuracy vs Computing Time (Seconds) on CMU-PIE(C IS THE NUMBER OF CLUSTERS)

	GraphSC					MODL		
C	**0.8**	**0.84**	**0.88**	**0.92**	**0.96**	**0.90**	**0.95**	**1**
4	12	N	13	N	14	N	11	12
20	32	38	57	72	105	24	26	N
36	48	69	73	82	123	51	52	N
52	N	65	97	127	N	44	46	N
68	71	87	130	218	N	60	63	N

IV. Experimental Results

In this section, we present experiments on image clustering. We compare the computation time and accuracy of our method (MODL) with GrapSC methods [14] [6]. We use data set [1] of CMU-PIE and COIL databases, examples are shown in fig. 1.

Instead of using PCA, K-SVD will be used for preprocessing, after getting the coefficient matrix ($\mathbf{A}$) by GraphSC and MODL, K-means will be used for clustering. We use computation time from matlab as efficiency evaluation metric, normalized mutual information as clustering accuracy evaluation metric [13] [6].

We also compared two manifold embedding methods LLE and Laplacian Embedding. Because of similar nature in calculation, we haven't seen noticeable difference in both computation efficiency and clustering accuracy.

All clustering tasks are based on a Windows 10 machine with Intel Core i7-2820M 2.3GHz CPU and 8GB RAM. All algorithms were implemented in MATLAB. We can easily find out from figure 2 and 3, MODL is more efficient than GraphSC in computing with same number of cluster. In table I and II, all clustering accuracy is fluctuating by ± 0.2, computation time is fluctuating by ± 10 seconds, "N" means none value.

V. Conclusion

In this paper, the authors present a novel modified online dictionary learning method to solve graph regularized sparse coding problem efficiently. Locally linear embedding method are also proposed to compare with original graph Laplacian constrain. During the processing, we found a preprocessing method can have big impact on the convergence time. With more optimized method like K-SVD for preprocessing, less time will be used for both GraphSC and MODL to converge. As graph regularized sparse coding can be represented as a quadratic convex problem plus a dictionary update problem, for very large dataset, interior point method [25] can be a great fit. Our further research would be solving large scale graph regularized sparse coding with interior point algorithm.

[1] http://www.cad.zju.edu.cn/home/dengcai/Data/MLData.html

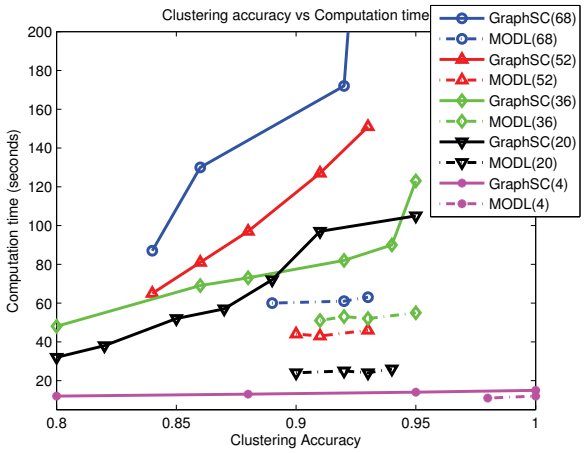


Fig. 2: CMU-PIE:Computing Time vs Accuracy of GraphSC and MODL (Number in parentheses is the number of Cluster used)

TABLE II: Clustering Accuracy vs Computing Time (Seconds) on COIL20 IMAGE LIBRARY(C IS THE NUMBER OF CLUSTERS)

	GraphSC					MODL		
C	**0.84**	**0.88**	**0.92**	**0.96**	**0.98**	**0.90**	**0.95**	**0.98**
4	N	17	N	21	27	N	16	19
8	54	87	132	179	N	31	33	N
12	N	102	187	241	N	49	53	N
16	94	147	211	N	N	63	71	N
20	N	184	289	N	N	77	81	N

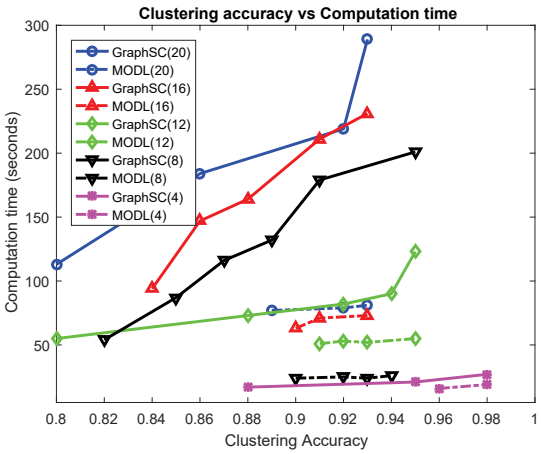


Fig. 3: COIL: Computing Time vs Accuracy of GraphSC and MODL (Number in parentheses is the number of Cluster used)

REFERENCES

[1] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," vol. 15, no. 12, pp. 3736–3745, 2006.

[2] J. Mairal, J. Mairal, M. Elad, M. Elad, G. Sapiro, and G. Sapiro, "Sparse representation for color image restoration," in *the IEEE Trans. on Image Processing*. ITIP, 2007, pp. 53–69.

[3] R. Grosse, R. Raina, H. Kwong, and A. Y. Ng, "Grosse et al. 149 shift-invariant sparse coding for audio classification."

[4] B. A. Olshausen, "Sparse coding of time-varying natural images," in *IN PROC. OF THE INT. CONF. ON INDEPENDENT COMPONENT ANALYSIS AND BLIND SOURCE SEPARATION*, 2000, pp. 603–608.

[5] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *in IEEE Conference on Computer Vision and Pattern Recognition(CVPR*, 2009.

[6] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, and D. Cai, "Graph regularized sparse coding for image representation," *IEEE Transactions on Image Processing*, pp. 1327–1336, 2011.

[7] D. J. F. B. A. Olshausen, "Sparse coding with an overcomplete basis set: a strategy employed by v1," *Vision Research*, vol. 37, pp. 3311–3325, 1997.

[8] A. J. Bell and T. J. Sejnowski, "The "independent components" of natural scenes are edge filters," 1997.

[9] S. Marelja, "Mathematical description of the responses of simple cortical cells," *Journal of the Optical Society of America*, vol. 70, pp. 1297–1300, 1980.

[10] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *In NIPS*. MIT Press, 2001, pp. 556–562.

[11] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Jour. of*, pp. 1457–1469, 2004.

[12] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *Journal of Computational and Graphical Statistics*, vol. 15, 2004.

[13] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized non-negative matrix factorization for data representation," *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, vol. 33, no. 8, pp. 1548–1560, 2011.

[14] S. Gao, I. W. hung Tsang, L. tien Chia, and P. Zhao, "Local features are not lonely laplacian sparse coding for image classification."

[15] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," 2010.

[16] L. K. Saul and S. T. Roweis, "An introduction to locally linear embedding," Tech. Rep., 2000.

[17] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing Systems 14*. MIT Press, 2001, pp. 585–591.

[18] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, pp. 267–288, 1994.

[19] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," 1995.

[20] Q. Gu and J. Zhou, "Gu, zhou: Neighborhood preserving nonnegative matrix factorization 1 neighborhood preserving nonnegative matrix factorization."

[21] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[22] D.P.Bertsekas, *Nonlinear Programming*. Athena Scientific Belmont, 1999.

[23] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Statist*, vol. 32, pp. 407–451, june 2004.

[24] R. L. B. E. Karl Sjostrand, Line H. Clemmensen, "Spasm: A matlab toolbox for sparse statistical modeling," *Journal of Statistical Software*, 2010.

[25] S. J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior point method for large scale l1 regularized least squares," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, Dec 2007.

VI. BIOGRAPHY

Lingdao Sha received his BS in electrical and computer engineering from Beijing University of Posts and Telecommunications (2011). He is now pursuing his PhD in electrical and computer engineering from University of Illinois at Chicago. His research interests are image processing, medical image processing and recognition, 3-D images, sparse coding and deep learning.

Dan Schonfeld received the B.S. degree in Electrical Engineering and Computer Science from the University of California at Berkeley, and the M.S. and Ph.D. degrees in Electrical and Computer Engineering from The Johns Hopkins University, in 1986, 1988, and 1990, respectively. In 1990, he joined the University of Illinois at Chicago, where he is currently a Professor in the Departments of Electrical and Computer Engineering, Computer Science, and Bioengineering. Dr. Schonfeld has been elevated to the rank of Fellow of the IEEE for contributions to image and video analysis. He was also elevated to the rank of Fellow of the SPIE for specific achievements in morphological image processing and video analysis. Dr. Schonfeld has been elected University Scholar of the University of Illinois and received the Graduate Mentoring Award of the University of Illinois at Chicago. He has authored over 200 technical papers in various journals and conferences. He was co-author of a paper that won the Best Paper Award at the ACM Multimedia Workshop on Advanced Video Streaming Techniques for Peer-to- Peer Networks and Social Networking 2010. He was also co-author of papers that won the Best Student Paper Awards in Visual Communication and Image Processing 2006 and IEEE International Conference on Image Processing 2006 and 2007. He is currently serving as Editor-in- Chief of the IEEE Transactions on Circuits and Systems for Video Technology. He has served as Deputy Editor-in- Chief of the IEEE Transactions on Circuits and Systems for Video Technology and Area Editor for Special Issues of the IEEE Signal Processing Magazine. His current research interests are in signal processing, image and video analysis, video retrieval and ommunications, multimedia systems, computer vision, medical imaging, and genomic signal processing.

Jing Wang received her B.S. degree in Mathematics from Shandong Normal University, the M.S. degree in Probability and Statistics from Beijing Normal University and the Ph.D. degree in Statistics from Michigan State University. In 2006, she joined the Departments of Mathematics, Statistics, and Computer Science at the University of Illinois at Chicago. Currently she is an Associate Professor in Statistics. Dr. Wang is an elected fellow in the International Statistical Institute. She is a faculty fellow in the Honors College at University of Illinois at Chicago. With her coauthors, she has published about 20 papers in peer-view journals. Her current research interests include semi-parametric and non-parametric regression, generalized additive models, variable and model selections, and dimension reduction