A Deep Graph Embedding Network Model for Face Recognition

Yufei Gan, Teng Yang, Chu He

Electronic Information School, Wuhan University, Wuhan 430072, China

Email: ganyufei@whu.edu.cn, tengyang@whu.edu.cn, chuhe@whu.edu.cn

Abstract—In this paper, we propose a new deep learning network "GENet", it combines the multi-layer network architecture and graph embedding framework. Firstly, we use simplest unsupervised learning PCA/LDA as first layer to generate the low-level feature. Secondly, many cascaded dimensionality reduction layers based on graph embedding framework are applied to GENet. Finally, a linear SVM classifier is used to classify dimension-reduced features. The experiments indicate that higher classification accuracy can be obtained by this algorithm on the CMU-PIE, ORL, Extended Yale B dataset.

Keywords—Deep Learning, Graph Embedding framework, Face Recognition.

I. INTRODUCTION

The task of image classification is fundamental for many computer vision tasks. Currently, many approaches have been proposed to solve this task, and many researches about image classification are related to dimensionality reduction.

In order to avoid the curse of dimensionality issue, many dimensionality reduction algorithms have been proposed, and Yan [1][2] has proposed a common framework based on the direct graph embedding to unify these algorithms. In the framework of graph embedding, we can use a unified view for understanding and explaining many of the popular dimensionality reduction algorithms, and a new dimensionality reduction algorithm called Marginal Fisher Analysis (MFA) has been proposed.

Recent research [3][4] shows the advantage of deep network in image classification, However, the framework of graph embedding do not have a multi-layer construction. A multi-layer construction can contribute the performance the of the single network.

GENet use deep network to enhance the performance of the framework of graph embedding. In GENet, a cascaded dimensionality reduction algorithm based on graph embedding network is applied to reduce feature dimensions.

Deep learning network mostly use an unsupervised learning as first layer and these unsupervised learning actually learn the feature from the data. The framework of graph embedding also learns the mapping that transforms high-dimensional representation into the desired low-dimensional representation.

Our contribution is that we proposed a simple and fast deep network for comparing and justifying other more advanced deep learning components or architectures like CNNs [5]. Once the parameters are fixed, training GENet is extremely simple and efficient, for the filter learning in GENet does not involve regularized parameters and does not require numerical optimization solver.

As the research further develops, researchers find the fact that the convolutional deep neural network (CNNs) has weak classification capacity in high-level layer [6] when compared to SVM. So SVM has been applied to replace the high-level layer recently, and our GENet also uses SVM as classifier.

II. GRAPH EMBEDDING FRAMEWORK

To represent each vertex of a graph as a low-dimensional vector, Yan [1] has proposed a general framework called graph embedding to offer a unified view for understanding and explaining many of the popular dimensionality reduction algorithms. In graph embedding, we use graph to describe the manifold structure of data, and we denote the sample set as $X = [x_1, x_2, \cdots, x_N], x_i \in \mathbb{R}^m$. In the supervised learning problem, the N_c class labels are assumed as $c_i \in \{1, 2, \cdots, N_c\}$ and denote π as the index set of samples belonging to class c. For an undirected weighted graph $G = \{X, W\}$ with vertex set X and similarity matrix $W \in \mathbb{R}^{N \times N}$. And in matrix W, each element of the real symmetric matrix W measures, for a pair of vertices, its similarity, which may be negative.

The diagonal matrix D and the Laplacian matrix L of a graph G are defined as

$$L = D - W, \ D_{ii} = \sum_{j \neq i} W_{ij}, \ \forall i. \tag{1}$$

We define an intrinsic graph to be the graph G itself and a penalty graph $G^p = \{X, W^p\}$ as a graph whose vertices X are the same as those of G, but whose edge weight matrix W^p corresponds to the similarity characteristics that are to be suppressed in the dimension-reduced feature space. we represent the low-dimensional representations of the vertices as a vector $y = [y_1, y_2, \cdots, y_N]^T$, where y_i is the low-dimensional representation of vertex x_i.

And for a dimensionality reduction problem, we require an intrinsic graph G and, optionally, a penalty graph G^p as input. Our graph-preserving criterion is given as follows:

$$y^* = \arg \min_{y^T By = d} \sum_{i \neq j} \|y_i - y_j\|^2 W_{ij} = \arg \min_{y^T By = d} y^T Ly,$$

$$\tag{2}$$

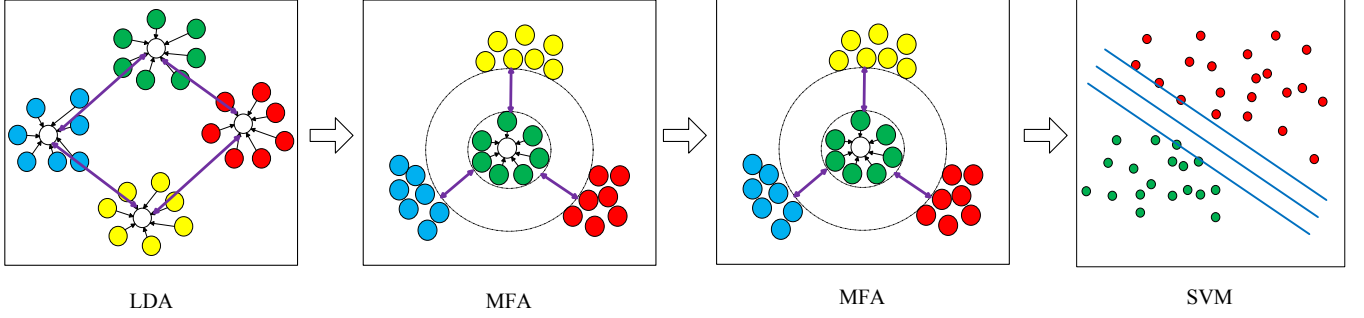where d is a constant and B is the constraint matrix ($B = L^p = D^p - W^p$)

Fig. 1. The construction of GENet, three cascaded dimensionality reduction algorithms based on the framework of graph embedding are shown in this picture. Different color dots represent different classes, hollow dots in the center of each kind of class represent the clustering center. We use unsupervised learning algorithms as first layer (for instance, in this figure, we select LDA). and two layers MFA as supervised learning layer. The classifier is SVM to efficiently classify the low-dimensional feature.

In the framework of graph embedding, we can describe the traditional dimensionality reduction algorithms in the same framework by setting intrinsic graph G and penalty graph G^p.

PCA [7] finds and removes the projection directions with maximal variance.To put it in another word, it finds and removes the projection direction with minimal variance:

$$
\begin{aligned}
w^* &= \arg \min_{w^T w = 1} w^T C w \quad with \\
C &= \frac{1}{N} \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T \\
&= \frac{1}{N} X(I - \frac{1}{N} e e^T) X^T
\end{aligned}
\tag{3}
$$

where $\bar{x}$ is the mean of all samples and e is a N dimensional vector with $e = [1, 1, \cdots, 1]^T$.

LDA [8] searches for the directions that are most effective for discrimination by minimizing the ratio between the intraclass and interclass scatters:

$$
\begin{aligned}
w^* &= \arg \min_{w^T S_B w = d} w^T S_W w = \arg \min_w \frac{w^T S_W w}{w^T S_B w} \\
&= \arg \min_w \frac{w^T S_W w}{w^T C w} \\
S_W &= \sum_{i=1}^{N} (x_i - \bar{x}^{c_i})(x_i - \bar{x}^{c_i})^T \\
&= X(I - \sum_{c=1}^{N_c} \frac{1}{n_c} e^c e^{cT}) X^T \\
S_B &= \sum_{c=1}^{N_c} n_c (\bar{x}^c - \bar{x})(\bar{x}^c - \bar{x})^T \\
&= N C - S_W
\end{aligned}
\tag{4}
$$

where $\bar{x}^c$ is the mean of the c-th class, and e^c is an N dimensional vector with $e^c(i) = 1, \ if \ c = l_i; 0, otherwise.$

In MFA (Marginal Fisher Analysis), the intrinsic graph characterizes the intraclass compactness and connects each data point with its neighboring points of the same class, while the penalty graph connects the marginal points and characterizes the interclass separability.

$$
\begin{aligned}
w^* &= \arg \min_w \frac{w^T X(D - W) X^T w}{w^T X(D^p - W^p) X^T w} \\
\tilde{S}_t &= \sum_i \sum_{i \in N_{k_1}^+(j) \ or \ j \in N_{k_1}^+(i)} \|w^T x_i - w^T x_j\|^2 \\
&= 2 w^T X(D - W) X^T w, \\
W_{i,j} &= \begin{cases} 1, & if \ i \in N_{k_1}^+(j) \ or \ j \in N_{k_1}^+(i) \\ 0, & else. \end{cases} \\
D_{ii}^w &= \sum_j W_{ij}^w. \\
\tilde{S}_p &= \sum_i \sum_{(i,j) \in P_{k_2}(c_i) \ or \ j \in P_{k_2}(c_j)} \|w^T x_i - w^T x_j\|^2 \\
&= 2 w^T X(D^P - W^P) X^T w, \\
W_{i,j}^p &= \begin{cases} 1, & if \ (i,j) \in P_{k_2}(c_i) \ or \ (i,j) \in P_{k_2}(c_j) \\ 0, & else. \end{cases} \\
D_{ii}^p &= \sum_j W_{ij}^p
\end{aligned}
\tag{5}
$$

where $N_{k_1}^+(i)$ indicates the index set of the k_1 nearest neighbors of the sample x_i in the class, and $P_{k_1}(c)$ is a set of data pairs that are the k_2 nearest pairs among the set $\{(i,j), i \in \pi_c, j \notin \pi_c\}$.

TABLE I.

The Common Graph Embedding View
for the Most Popular Dimensionality Reduction Algorithms

Algorithm	W&B Definition
PCA/KPCA/2DPCA	$W_{ij} = \frac{1}{N}, i \neq j; B = I$
LDA/KDA/2DLDA/DATER	$W_{ij} = \delta_{c_i, c_j}/n_c; B = I - \frac{1}{N} e e^T$

III. GENET

In the Graph Embedding, some dimensionality reduction algorithms can be applied to our GENet, such as PCA, LDA, IOSMAP, LLE, and so on. In order to research the advantage of the deep network construction, we simply use the simplest dimensionality reduction linear algorithms–PCA and LDA. Moreover, in consideration of that Marginal Fisher Analysis

Fig. 2. The sample images cropped from the face database Extended Yale B (first row), ORL (second row), PIE_32x32 (third row), and Pose05_64x64 (fourth row), respectively.

show an competitive performance in Yan's [1] paper, we also introduce liner MFA to our GENet.

In our GENet, we use unsupervised learning algorithm as the first layer to obtain sufficient low-level-feature. In the experiment section, we will find the fact that use unsupervised learning algorithm as the first layer perform better than supervised learning algorithm.

The construction of GENet is shown in Figure 1. In the figure, three cascaded dimensionality reduction algorithms based on the framework of graph embedding are shown in this picture. Different color dots represent different classes, hollow dots in the center of each kind of class represent the clustering center.

IV. EXPERIMENT

In our experiments, we use the Extended Yale Face Database B, CMU PIE, and ORL databases for face recognition to evaluate our GENet.

With the following reasons, we decide to use the dengcai versions [9], [10], [11], [12] of Extended Yale Face Database B, CMU PIE and ORL face sets:

A. Faces are already standardized according to eye locations – so that when we compare the performances of identification algorithms, we do not need to worry if the standardization approaches are the same.

B. There are two versions, 32×32 pixels and 64×64 pixels, available for each set, so that we can see if our approach works for different sizes.

C. The above UIUC site provides many holistic algorithms, including newly developed ones, in source codes, they also provided best results of many algorithms (5 to 10 algorithms) for 32×32 versions of above data sets.

In the dengcai versions of Extended Yale Face Database B, PIE and ORL face sets, the Extended Yale Face Database B has 38 individuals and around 64 near frontal images

under different illuminations per individual, the data file – YaleB_32x32 contains 2,414 images which are cropped and resize to 32x32 pixels. And CMU PIE contains 41,368 images of 68 people, each person under 13 different poses, 43 different illumination conditions, and with 4 different expressions, the CMU PIE data file – PIE_32x32 contains 11,554 images which are cropped and resize to 32x32 pixels. The ORL face sets have ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement), and the data file – ORL_32x32 contains 400 images which are cropped and resize to 32x32 pixels.

The implementation of the PCA/LDA/MFA are obtained from dengcai's personal homepage: http://www.cad.zju.edu.cn/home/dengcai/Data/DimensionReduction.html

The experiment on ORL use five different training data size (from 1 to 5) to train GENet, and the remaining images are used for testing. However in PIE dataset and Extended Yale B dataset, the number of images of each person are different, so use the same test data size and the remaining images are used for training.

We use PCA+MFA+PCA+MFA (100, 70, 60, 40), PCA+MFA (100, 40), LDA+MFA+LDA+MFA (100, 70, 60, 40), LDA+MFA (100, 40), LDA+MFA+PCA+MFA (100, 70, 60, 40), PCA+MFA+LDA+MFA (100, 70, 60, 40), PCA+MFA+MFA (100, 70, 40), LDA+MFA+MFA (100, 70, 40) these combinations, the the numbers in parentheses are the corresponding feature dimensions, and the parameters of MFA are set to $k_1 = 10, k_2 = 500$ in the experiment on ORL, $k_1 = 2, k_2 = 440$ in the experiment on PIE, and $k_1 = 10, k_2 = 500$ in the experiment on Extended Yale B. The results are shown in Table II, Table III and Table IV

Comparing the experiment on Extended Yale B with the experiment on PIE, We can know that LDA algorithm have

TABLE II.

Face Recognition Accuracies of PCA+MFA+PCA+MFA , PCA+MFA,
LDA+MFA+LDA+MFA, LDA+MFA, LDA+MFA+PCA+MFA,
PCA+MFA+LDA+MFA, PCA+MFA+MFA, LDA+MFA+MFA on the ORL Database.
(numbers in parentheses are the corresponding feature dimensions)

Train/Test	1/9	2/8	3/7	4/6	5/5
PCA+MFA+PCA+MFA	65.00%	80.00%	75.00%	89.16%	96.50%
PCA+MFA [1]	73.61%	80.31%	83.21%	88.75%	94.00%
LDA+MFA+LDA+MFA	70.27%	77.81%	72.85%	72.50%	69.00%
LDA+MFA	64.72%	79.06%	72.14%	68.75%	76.50%
LDA+MFA+PCA+MFA	65.55%	74.37%	79.28%	72.91%	70.00%
PCA+MFA+LDA+MFA	67.77%	77.18%	80.00%	**92.91**%	70.00%
PCA+MFA+MFA	**74.16**%	**80.62**%	**88.21**%	88.33%	**97.50**%
LDA+MFA+MFA	67.50%	75.93%	71.78%	69.16%	95.50%

Note that the approach PCA+MFA is proposed in [1].

TABLE III.

Face Recognition Accuracies of PCA+MFA+PCA+MFA , PCA+MFA,
LDA+MFA+LDA+MFA, LDA+MFA, LDA+MFA+PCA+MFA,
PCA+MFA+LDA+MFA, PCA+MFA+MFA, LDA+MFA+MFA on the Pose05_64x64
Database. (numbers in parentheses are the corresponding feature dimensions)

Test	30	20	10
PCA+MFA+PCA+MFA	**79.60**%	**84.92**%	**91.61**%
PCA+MFA [1]	54.26%	82.35%	80.88%
LDA+MFA+LDA+MFA	2.64%	1.47%	6.02%
LDA+MFA	2.64%	1.47%	6.02%
LDA+MFA+PCA+MFA	2.74%	3.01%	6.02%
PCA+MFA+LDA+MFA	72.20%	79.55%	82.35%
PCA+MFA+MFA	59.01%	81.25%	90.29%
LDA+MFA+MFA	0.78%	2.79%	2.35%

Note that the approach PCA+MFA is proposed in [1].

TABLE IV.

Face Recognition Accuracies of PCA+MFA+PCA+MFA , PCA+MFA,
LDA+MFA+LDA+MFA, LDA+MFA, LDA+MFA+PCA+MFA,
PCA+MFA+LDA+MFA, PCA+MFA+MFA, LDA+MFA+MFA on the YaleB_32x32
Database. (numbers in parentheses are the corresponding feature dimensions, and the
number of test images is 1)

PCA+MFA+PCA+MFA	61.05%	PCA+MFA [1]	73.68%
LDA+MFA+LDA+MFA	82.10%	LDA+MFA	82.10%
LDA+MFA+PCA+MFA	76.84%	PCA+MFA+LDA+MFA	73.68%
PCA+MFA+MFA	62.10%	LDA+MFA+MFA	**84.21**%

Note that the approach PCA+MFA is proposed in [1].

not a good performance on PIE dataset, but it have a good performance on Extended Yale B dataset, the results may due to limitations of data distribution assumptions.

Another observation is a obvious result – in most cases, the GENet with multi-layer performance better than GENet with single-layer.

V. CONCLUSION

In this paper, we proposed a deep learning network based on the graph embedding. GENet combine the deep network and the framework of graph embedding, it uses a multi-layer graph embedding construction to classify images. Using GENet to compute filters does not require numerical optimization solver so the training process can be extremely efficient. Because of the of framework of the graph embedding, GENet can be treated as the cascaded network, the different of each layer is the setting of intrinsic graph and penalty graph.

Our results indicate that GENet can perform fast and accuracy in face recognition datasets such as Extended Yale B, CMU PIE, and ORL databases. Moreover, the results show the fact that multi-layer construction can perform better than single-layer construction. GENet makes it possible to find a adaptive algorithm to reduce dimensions and try to avoid the impact of data distribution assumptions.

In feature work, we hope to apply the extensions of graph embedding – kernel and tensor to our GENet, and we hope to find the effective method to let the parameters of GENet, specially the algorithm of each layer, can learn from the data.

REFERENCES

[1] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: a general framework for dimensionality reduction," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 1, pp. 40–51, 2007.

[2] S. Yan, D. Xu, B. Zhang, and H.-J. Zhang, "Graph embedding: A general framework for dimensionality reduction," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 830–837.

[3] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "Pcanet: A simple deep learning baseline for image classification?" *arXiv preprint arXiv:1404.3606*, 2014.

[4] Y. Bengio, "Learning deep architectures for ai," *Foundations and trendstextregistered in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[6] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," *arXiv preprint arXiv:1403.6382*, 2014.

[7] I. Jolliffe, *Principal component analysis*. Wiley Online Library, 2005.

[8] A. M. Mart'inez and A. C. Kak, "Pca versus lda," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 2, pp. 228–233, 2001.

[9] D. Cai, X. He, Y. Hu, J. Han, and T. Huang, "Learning a spatially smooth subspace for face recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition Machine Learning (CVPR'07)*, 2007.

[10] D. Cai, X. He, and J. Han, "Spectral regression for efficient regularized subspace learning," in *Proc. Int. Conf. Computer Vision (ICCV'07)*, 2007.

[11] D. Cai, X. He, J. Han, and H.-J. Zhang, "Orthogonal laplacianfaces for face recognition," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3608–3614, 2006.

[12] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using laplacianfaces," *IEEE Trans. Pattern Anal. Mach. Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.