Sparse Similarity Metric Learning for Kinship Verification

Yuan Fang[1], Yan Yan[1*], Si Chen[2], Hanzi Wang[1], Chang Shu[3]

[1]*Fujian Key laboratory of Sensing and Computing for Smart City, School of Information Science and Technology, Xiamen University, Xiamen 361005, China*

[2]*School of Computer and Information Engineering, Xiamen University of Technology, Xiamen 361024, China*

[3]*School of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China*

fangyuan_xmu@163.com, yanyan@xmu.edu.cn, chensi@xmut.edu.cn
hanzi.wang@xmu.edu.cn, changshu@uestc.edu.cn

***Abstract*—Metric learning technique learns a linear transformation of the given training data which can significantly promote the performance of a prediction task, such as kinship verification. However, many of the existing metric learning methods do not explicitly regularize for sparsity or low-rank, which in practice usually results in high-rank solutions that are not only time-consuming but also tend to overfitting. In addition, some methods simply neglect the positive semidefinite (PSD) constraint, causing the learned metric to be potentially noisy. In this paper, we propose an effective sparse similarity metric learning (SSML) method which enforces both the group sparsity and the PSD constraints on the learned similarity matrix for kinship verification. In order to solve the proposed optimization problem efficiently, we successfully apply the alternating direction method of multipliers (ADMM) to obtain the optimal solution. Experimental results demonstrate that the proposed method achieves competitive results compared with other state-of-the-art metric learning methods on widely used kinship datasets.**

***Index Terms*—Sparse; Similarity learning; Positive semidefinite constraint; Alternating direction method of multipliers; Kinship verification**

I. Introduction

Metric learning has been extensively researched for decades. The reason for its popularity can be attributed to the fact that many computer vision applications heavily depend on the use of similarity or distance metrics. Among them, kinship verification is an important application. The task of kinship verification [1] is to determine whether there is a kinship relationship between a given facial image pair of two persons. In practice, this task is very challenging because the facial images are usually taken under uncontrolled environments and suffer from numerous variations including different lighting conditions, changes in facial expression, pose and age (see Fig. 1 for an illustration). Besides, the standard protocol for evaluating kinship verification defines the exclusive person identities in the training sets and test sets, thus requiring the good generalization capability of the metric learning methods.

There are two major components of the kinship verification task: 1) the descriptors employed to represent the images; 2) the similarity or distance metric applied to compare the descriptors. In this paper, we focus on the latter.

(a) (b) (c) (d)

Fig. 1. An illustration of several cropped and aligned facial images from the KinFaceW-I and KinFaceW-II datasets. Facial images from the first row is collected from the KinFaceW-I dataset while those from the second row is collected from the KinFaceW-II dataset. (a) Facial images with the father-son (F-S) kinship relationship. (b) Facial images with the father-daughter (F-D) kinship relationship. (c) Facial images with the mother-son (M-S) kinship relationship. (d) Facial images with the mother-daughter (M-D) kinship relationship.

A suitable metric can significantly boost the performance of a pattern recognition system. In order to obtain a good quality metric, metric learning algorithms attempt to learn a similarity (or distance) function that can yield large similarities (or small distances) between similar pairs of samples, while providing small similarities (or large distances) between dissimilar pairs of samples. Recently, there has been considerable interest in learning an appropriate similarity or distance metric. Among them, Large Margin Nearest Neighbor (LMNN) [2], Information Theoretic Metric Learning (ITML) [3] and Logistic Discriminant Metric Learning (LDML) [4] are three representative approaches. LMNN [2] learns a Mahalanobis metric to improve the performance of the k-nn classification, where the goal is to force the k nearest neighbors of each example to share the same class label, while separating examples with different class labels by a large margin. ITML [3] uses an information theoretic method to obtain a Mahalanobis distance function, which is constrained under large-scale constraints and regularized by a known prior matrix. LDML [4] employs the logistic discriminant function to model the posteriori class probability of elements belonging to the same class.

Although existing metric learning methods have reported good performance across various tasks, most of them still encounter the following problems. Firstly, some metric learn-

* Corresponding author

978-1-5090-5316-2/16/$31.00 © 2016 IEEE

ing methods, such as LMNN [2] and Pairwise Constrained Component Analysis (PCCA) [5], learn the distance metric without any regularization and often become prone to overfitting. Some work, including ITML [3], applies an explicit regularizer. However, it learns the high-rank distance matrix, which cannot effectively suppress the data with noise and is subject to overfitting. Secondly, many approaches ignore the PSD constraint (which can effectively smooth the learned metric) and learn a potentially noisy metric. Several work such as [5] ensures the PSD constraint by factorizing the metric M as $M = L^T L$. However, the optimization problem with regard to L is not convex, thus making the final solution sub-optimal.

To address the above issues, we propose a sparse similarity metric learning method, which imposes the group sparsity and the PSD constraints on the learned metric to avoid overfitting. Besides, to solve the proposed optimization problem, we derive an efficient learning algorithm depend on the alternating direction method of multiplier (ADMM) [6] to seek an optimal solution. The key characteristic of ADMM [6] lies in that it does not require the projection onto the PSD cone at each iteration, and thus effectively improve the efficiency of the proposed optimization function. A series of experiments on kinship verification demonstrate the efficiency and effectiveness of the proposed approach.

Notations: For matrices A, B, $\langle A, B\rangle = tr(A^T B)$, where $tr(\cdot)$ denotes the trace of a matrix. Let $\mathbb{S}^d$ and $\mathbb{S}_+^d$ denote the sets of $d \times d$ real symmetric and real symmetric PSD matrices, respectively. Let $\Pi_{\mathbb{S}_+^d}(A)$ indicates the orthogonal projection of the matrix $A \in \mathbb{S}^d$ onto the PSD cone $\mathbb{S}_+^d$. For matrix A, we denote $A_{i\cdot}$ as the ith row of A, and $\|A\|_F$ as the Frobenius norm on A. For $a \in \mathbb{R}$, $[a]_+ = \max(a, 0)$. Given a set of points X, where each point is represented as a vector $x \in \mathbb{R}^d$. S and D are denoted as the sets of similar pairs and dissimilar pairs, respectively, and $y_{ij} \in \{1, -1\}$ indicates whether a pair (x_i, x_j) belongs to the same class or not.

II. Sparse Similarity Metric Learning

This section introduces our proposed method, the Sparse Similarity Metric Learning (SSML) method in detail.

A. Data Preprocessing

Inspired by [7], we first employ Principal Component Analysis (PCA) to remove the noise, and then map the original data points onto the intra-class subspace to reduce the influence of large intra-class variations. The intra-class covariance matrix is defined as,

$$G = \sum_{(x_i, x_j) \in S} (x_i - x_j)(x_i - x_j)^T \ . \tag{1}$$

Based on the eigendecomposition of G, we can obtain $\Lambda = (\lambda_1, ..., \lambda_t)$ and $P = (v_1, ..., v_t)$, which are the top leading t eigenvalues and the corresponding eigenvectors of G, respectively, where t is the dimensional of data after PCA. The mapped data points are then defined as $x' = \text{diag}(\lambda_1^{-\frac{1}{2}}, ..., \lambda_t^{-\frac{1}{2}}) P^T x$, where $\text{diag}(\cdot)$ denotes the diagonalization of a vector. In order to simplify the notation, we mention x in the rest of this paper which in fact refers to x'.

B. Sparse Similarity Metric Learning

Our goal is to learn a similarity function, i.e.,

$$S_M(x_i, x_j) = x_i^T M x_j \tag{2}$$

to measure the similarity between a pair of samples, where $M \in \mathbb{S}_+^d$. To learn such a metric, we take a logistic loss function into account as follows,

$$f_M(x_i, x_j) = \frac{1}{\delta} \log(1 + e^{\delta y_{ij}(\mu - S_M(x_i, x_j))}) \ , \tag{3}$$

where δ is the sharpness parameter; μ is the mean similarity among all the pairs used in the training data, which is used considering that the similarity function has a lower bound. The logistic function is known to be smooth and convex, and can also provide a soft margin to separate the two classes.

Hence, the overall loss function is formulated as

$$F(M) = \sum_{(x_i, x_j) \in S} f_M(x_i, x_j) + \alpha \sum_{(x_i, x_j) \in D} f_M(x_i, x_j) \ , \tag{4}$$

where α is a parameter to balance the contribution between the positive pairs and the negative pairs.

Additionally, we aim to learn a metric M that depends only on the informative features. More precisely, if the input dimension is non-informative, the corresponding row of M should suppress the feature. In other words, a natural row grouping of the entries of M is suggested to enforce the sparsity. As a result, we boost the row-sparsity by imposing the mixed-norm regularization

$$\|M\|_{2,1} = \sum_{i=1}^{d} \|M_{i\cdot}\|_2 \ , \tag{5}$$

As a result, we formulate the proposed Sparse Similarity Metric Learning (SSML) as the following optimization problem

$$\min_{M \in \mathbb{S}_+^d} F(M) + \gamma \|M\|_{2,1} \ , \tag{6}$$

where γ is the regularization parameter.

III. Alternating Direction Method of Multipliers

At the first glance, the optimization problem (6) can be solved by first learning a metric M without the PSD constraint via a gradient descent solver, and then projecting M onto $\mathbb{S}_+^d$ after each gradient step. However, this option is computationally expensive due to the projection onto the PSD cone is required at each iteration. Therefore, in this paper we consider the ADMM [6] to efficiently optimize (6).

We first adapt (6) in the following,

$$\min_{M,W,Z} F(M) + g(W) + h(Z) \quad s.t.\ M = W = Z \ , \tag{7}$$

where

$$g(W) = \gamma \|W\|_{2,1} \ , \ \ h(Z) = \begin{cases} 0 & Z \in \mathbb{S}_+^d \\ +\infty & Z \notin \mathbb{S}_+^d \end{cases} \ . \tag{8}$$

We define the augmented Lagrangian as

$$\begin{aligned}\mathcal{L}(M,W,Z,\Lambda_M,\Lambda_W) &= F(M)+g(W)+h(Z)\\ &+\langle\Lambda_M, M-Z\rangle+\frac{\rho}{2}\|M-Z\|_F^2\\ &+\langle\Lambda_W, W-Z\rangle+\frac{\rho}{2}\|W-Z\|_F^2\,,\end{aligned} \tag{9}$$

where $\Lambda_M, \Lambda_W \in \mathbb{S}^d$ are the Lagrange multiplies, and $\rho > 0$ is a scaling parameter. Therefore, the scaled form of ADMM for this problem can be written as

$$M^{k+1} \leftarrow \arg\min_{M\in\mathbb{S}^d} F(M)+\frac{\rho}{2}\|M-(Z^k-U_M^k)\|_F^2\,, \tag{10}$$

$$W^{k+1} \leftarrow \arg\min_{W\in\mathbb{S}^d} g(W)+\frac{\rho}{2}\|W-(Z^k-U_W^k)\|_F^2\,, \tag{11}$$

$$\begin{aligned}Z^{k+1} \leftarrow \arg\min_{Z\in\mathbb{S}^d} & h(Z)\\ &+\rho\|Z-\frac{1}{2}(M^{k+1}+W^{k+1}+U_M^k+U_W^k)\|_F^2\,,\end{aligned} \tag{12}$$

$$U_M^{k+1} \leftarrow U_M^k + M^{k+1} - Z^{k+1}\,, \tag{13}$$

$$U_W^{k+1} \leftarrow U_W^k + W^{k+1} - Z^{k+1}\,, \tag{14}$$

where $U_M = \frac{1}{\rho}\Lambda_M$, $U_W = \frac{1}{\rho}\Lambda_W$.

The M-update (10) is an unconstrained convex optimization problem and $F(M)$ is smooth. Therefore, the optimization problem (10) can be solved by using any standard method, such as Newton's method or a quasi-Newton method.

We update the optimization problem (11) by first removing the symmetry constraint, and then projecting W^{k+1} onto $\mathbb{S}^d$. According to [8], let $V = Z^k - U_W^k$, the solution of (11) without the symmetry constraint can be given by an element-wise threshold operation

$$W_{ij}^{k+1} = V_{ij}[1-\frac{\gamma}{\rho\|V_{i\cdot}\|_2}]_+\,. \tag{15}$$

In Z-update (12), we simply obtain the Z^{k+1} via the orthogonal projection

$$Z^{k+1} \leftarrow \Pi_{\mathbb{S}_+^d}[\frac{1}{2}(M^{k+1}+W^{k+1}+U_M^k+U_W^k)]. \tag{16}$$

The optimization problem (7) is solved by cyclic updating M, W, Z, U_M and U_W until convergence. We summarize the proposed ADMM based method in Algorithm 1.

Algorithm 1 Sparse Similarity Metric Learning via ADMM

Input: X: input $d\times n$ matrix, S: set of similar pairs, D: set of dissimilar pairs, δ: sharpness parameter, γ: regularization parameter, ρ: scaling parameter

Output: M: the learned similarity metric matrix

1: $k=1$; $\alpha=\frac{|S|}{|D|}$; initialize $M^k=W^k=Z^k$ as an identity matrix; $U_M^k \leftarrow 0$; $U_W^k \leftarrow 0$;
2: **repeat**
3: Obtain M^{k+1} by solving optimization problem (10);
4: Obtain W^{k+1} by solving optimization problem (11);
5: Obtain Z^{k+1} using Eq. (16);
6: Update U_M^{k+1} using Eq. (13);
7: Update U_W^{k+1} using Eq. (14);
8: $k=k+1$;
9: **until** Convergence criterion is satisfied
10: **return** M^k

IV. Experiments

To evaluate the proposed method, we conduct extensive kinship verification experiments on two publicly available kinship datasets, i.e., Kin Faces in the Wild I (KinFaceW-I) [1] and Kin Faces in the Wild II (KinFaceW-II) [1].

A. dataset

The KinFaceW-I [1] and KinFaceW-II [1] are two challenging kinship verification datasets, which have been widely used for benchmark evaluation. The difference between the two datasets is that the kinship facial images of each pair in KinFaceW-I are obtained from different pictures while those in KinFaceW-II are acquired from the same picture. The facial images in the two datasets contain large variations in pose, background, illumination, ethnicity, age and expression.

Both KinFaceW-I and KinFaceW-II datasets have four kinship relationships: father-son (F-S), father-daughter (F-D), mother-son (M-S) and mother-daughter (M-D). KinFaceW-II is larger than KinFaceW-I in size. While KinFaceW-I contains 156, 134, 116 and 127 pairs of facial images for these four kinship relationships, respectively. Each kinship relationship contains 250 pairs of facial images for each kinship relationship in the KinFaceW-II. Some examples collected from KinFaceW-I and KinFaceW-II datasets are shown in Fig. 1.

B. Experimental settings

For our experiments, facial images are cropped and aligned into 64×64 pixels for each dataset based on the provided eyes positions. In this paper, we focus on the image-restricted settings where only the kinship relationship information is available in the training split. We follow the benchmark provided in [1], where the datasets are equally split into five cross validation folds.

In our experiment, we take representative descriptors developed by Lu et al. [9], that is, Local Binary Patterns (LBP) [10] and Histogram of Gradients (HOG) [11]. For LBP, each facial image is separated into 8×8 non-overlapping blocks with the size of 8×8. For each block, a 59-dimensional uniform LBP feature is extracted, and then concatenated them to a 3776-dimensional descriptor. For HOG, each facial image is first split into 16×16 non-overlapping blocks with the size of 4×4, and then split into 8×8 non-overlapping blocks with the size of 8×8. For each block, a 9-dimensional HOG feature is extracted, and then concatenated them to a 2880-dimensional descriptor.

We compare the proposed method with five representative metric learning approaches, including Large Margin Nearest Neighbor (LMNN) [2], Information Theoretic Metric Learning (ITML) [3], Neighborhood Repulsed Metric Learning (NRML) [1], Logistic Discriminant Metric Learning (LDML) [4] and Generalized Sparse Metric Learning (GSML) [12]. Note that for LMNN, NRML and GSML, we learn distance metrics in the image-unrestricted settings since they learn the

TABLE I
VERIFICATION ACCURACY (%) OF VARIOUS METHODS ON DIFFERENT SUBSETS OF THE KINFACE-I DATASET

Method	Feature	F-S	F-D	M-S	M-D	Mean
NRML [1]	LBP	81.43	69.76	67.23	72.87	72.82
	HOG	83.68	74.64	71.56	79.96	77.46
ITML [3]	LBP	79.49	71.28	69.84	70.10	72.68
	HOG	82.40	75.01	72.83	78.73	77.24
LMNN [2]	LBP	76.92	69.42	69.82	67.76	70.98
	HOG	79.82	75.00	75.40	76.39	76.65
GSML [12]	LBP	75.01	67.17	66.81	68.87	69.47
	HOG	76.95	**76.55**	71.58	76.81	75.47
LDML [4]	LBP	78.20	70.17	69.37	70.50	72.06
	HOG	78.21	75.75	74.53	77.19	76.42
SSML	LBP	**81.74**	**75.36**	**71.49**	**77.93**	**76.63**
	HOG	**84.63**	75.00	**76.27**	**82.30**	**79.55**

TABLE II
VERIFICATION ACCURACY (%) OF VARIOUS METHODS ON DIFFERENT SUBSETS OF THE KINFACE-II DATASET

Method	Feature	F-S	F-D	M-S	M-D	Mean
NRML [1]	LBP	79.20	71.60	72.20	68.40	72.85
	HOG	80.80	72.80	74.80	70.40	74.70
ITML [3]	LBP	77.00	73.00	69.80	69.60	72.35
	HOG	81.00	74.60	75.40	73.20	76.05
LMNN [2]	LBP	77.80	73.20	70.60	70.40	73.00
	HOG	83.20	75.60	77.60	77.40	78.45
GSML [12]	LBP	75.60	72.00	69.20	71.80	72.15
	HOG	83.38	75.20	75.80	76.40	77.70
LDML [4]	LBP	77.40	74.40	71.40	71.20	73.60
	HOG	80.00	71.80	72.80	71.60	74.05
SSML	LBP	**82.40**	**78.60**	**79.80**	**77.93**	**79.68**
	HOG	**85.00**	**77.00**	**80.40**	**78.40**	**80.15**

metric under supervision. Our method contains three parameters, i.e., the balance parameter α, the sharpness parameter δ and the regularization parameter γ. We simply set α to $\frac{|S|}{|D|}$, and estimate δ and γ using cross validation.

C. Results and Analysis

Table I and Table II summarize the verification results of different metric learning methods on the KinFaceW-I and KinFaceW-II datasets, respectively. Although LMNN, NRML and GSML employ the additional information (these methods require the strong supervised learning while our method only utilizes the kinship relationship information), it can be seen that our approach obviously outperforms the state-of-the-art methods on the four kinship relationships except the F-D with the HOG descriptor on KinFaceW-I (which is only slightly worse than GSML).

The success of the proposed method is mainly due to the sparse restriction, the PSD constraint and the soft-margin loss function. There are three important points to be highlighted. First, GSML also imposes the sparsity regularization, but its performance is not as good as our method. Second, LMNN can also give the sparse metric, however, our method outperforms it. This is because LMNN lacks of regularization, and tends to overfitting. Third, although LDML also uses a logistic loss function to learn a metric, the propose method achieves better performance. The reason is that LDML ignores the PSD constraint, and learns a free distance function, thus making the learned metric potentially noisy.

V. CONCLUSION

In this paper, we have presented a new metric learning method, named Sparse Similarity Metric Learning (SSML). The proposed method restricts the learned similarity metric to be both PSD and group sparsity, and thus can effectively deal with the high dimensional and noisy data. Furthermore, we have proposed an adaptation of ADMM to solve the PSD constraint based optimization problem efficiently. We have conducted experiments on two kinship verification datasets to evaluate the performance of the proposed method, and shown its effectiveness compared with several state-of-the-art methods.

ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China under Grants 61571379, 61472334, and 61503315, and also supported by the Fundamental Research Funds for the Central Universities under Grants 20720162012.

REFERENCES

[1] J. Lu, X. Zhou, Y. P. Tan, Y. Shang, and J. Zhou, "Neighborhood repulsed metric learning for kinship verification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 2, pp. 331–345, 2014.

[2] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Proceedings of the 2005 Advances in neural information processing systems*, 2005, pp. 1473–1480.

[3] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 209–216.

[4] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? metric learning approaches for face identification," in *Proceedings of the 12th IEEE Conference on International Conference on Computer Vision*. IEEE, 2009, pp. 498–505.

[5] A. Mignon and F. Jurie, "Pcca: A new approach for distance learning from sparse pairwise constraints," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2666–2672.

[6] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[7] Q. Cao, Y. Ying, and P. Li, "Similarity metric learning for face recognition," in *Proceedings of the 14th IEEE Conference on International Conference on Computer Vision*. IEEE, 2013, pp. 2408–2415.

[8] M. Kowalski, "Sparse regression using mixed norms," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 303–324, 2009.

[9] J. Lu, J. Hu, V. E. Liong, X. Zhou, A. Bottino, I. Ul Islam, T. Figueiredo Vieira, X. Qin, X. Tan, S. Chen *et al.*, "The fg 2015 kinship verification in the wild evaluation," in *Proceedings of the 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition*, vol. 1. IEEE, 2015, pp. 1–7.

[10] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2005, pp. 886–893.

[12] K. Huang, Y. Ying, and C. Campbell, "Gsml: A unified framework for sparse metric learning," in *Proceedings of the 9th IEEE International Conference on Data Mining*. IEEE, 2009, pp. 189–198.