Weighted Graph Embedding-Based Metric Learning for Kinship Verification

Jianqing Liang, Qinghua Hu, *Senior Member, IEEE*, Chuangyin Dang, *Senior Member, IEEE*, and Wangmeng Zuo, *Senior Member, IEEE*

***Abstract*— Given a group photograph, it is interesting and useful to judge whether the characters in it share specific kinship relation, such as father–daughter, father–son, mother–daughter, or mother–son. Recently, facial image-based kinship verification has attracted wide attention in computer vision. Some metric learning algorithms have been developed for improving kinship verification. However, most of the existing algorithms ignore fusing multiple feature representations and utilizing kernel techniques. In this paper, we develop a novel weighted graph embedding-based metric learning (WGEML) framework for kinship verification. Inspired by the fact that family members usually show high similarity in facial features like eyes, noses, and mouths, despite their diversity, we jointly learn multiple metrics by constructing an intrinsic graph and two penalty graphs to characterize the intraclass compactness and interclass separability for each feature representation, respectively, so that both the consistency and complementarity among multiple features can be fully exploited. Meanwhile, combination weights are determined through a weighted graph embedding framework. Furthermore, we present a kernelized version of WGEML to tackle nonlinear problems. Experimental results demonstrate both the effectiveness and efficiency of our proposed methods.**

***Index Terms*— Weighted graph embedding, metric learning, kinship verification.**

I. Introduction

RECENT evidences in visual signal processing indicate that human appearance may provide valuable clues for biological relationship prediction [1], [2], which stimulates efforts in kinship verification from facial images over the last few years [3]–[7]. Kinship verification via facial image analysis is an emerging and interesting task in computer vision. It aims at distinguishing parent-offspring relations by measuring similarities between the facial appearance. Its practical applications include a variety of fields, such as finding missing children [8], fugitive identity confirmation and social media analysis [9].

Besides its extensive applications, kinship study using images is inspired by understanding single face image, an essential part of face recognition. Face recognition is an active research topic in the field of computer vision, and thus has received a great deal of attention in the past few decades [10]–[18]. The difficulties of face recognition stem from the subtle differences between faces and numerous impact factors on facial appearance [19]. As a special and hard case of object recognition problem, there are two crucial stages in a face recognition system, i.e., face representation and face matching. Face representation is to extract discriminative features to separate face images better, while face matching aims to design effective models to distinguish different face images. In general, face recognition can be categorized into two primary tasks: face verification and face identification. The former tries to ascertain whether a pair of face images is from the same person or not. The latter attempts to recognize the identity of a person by comparing the probe face image with a set of gallery face images with known identities.

Kinship verification shares certain characteristics with conventional face verification, as both of them are confronted with similar sources of variation in facial appearance, which can be classified into intrinsic factors (e.g., facial expression, age, hair, glasses, etc.) and extrinsic factors (e.g., pose, scale, illumination, etc.) [19]. Intrinsic factors are purely due to the essential attributes of the face while extrinsic factors are caused by interaction with the observer. Hence, face representation and face matching are of vital importance to both of these two tasks. Different from face verification which focuses on relation between different face images of an entity, kinship verification pays attention to relation between multiple entities, making it a more challenging task. The difficulties can be summarized in the following aspects [55]. First, compared with face verification, the complex relation among multiple entities puts forward a higher demand to feature representation. Second, it is practically infeasible to collect a large number of face images with kin relations. Therefore, how to make full use of limited training samples and effectively solve the small sample size problem poses a great challenge to current learning algorithms. Last but not the least, despite the

Manuscript received March 18, 2018; revised August 27, 2018; accepted October 2, 2018. Date of publication October 10, 2018; date of current version November 2, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61732011, Grant 61432011, Grant U1435212, and Grant 61502332 and in part by the Government of Hong Kong SAR under Grant CityU 11301014. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Peter Tay. *(Corresponding author: Qinghua Hu.)*

J. Liang and Q. Hu are with the College of Intelligence and Computing, Tianjin University, Tianjin 300350, China (e-mail: liangjianqing@tju.edu.cn; huqinghua@tju.edu.cn).

C. Dang is with the Department of Systems Engineering and Engineering Management, City University of Hong Kong, Hong Kong (e-mail: mecdang@cityu.edu.hk).

W. Zuo is with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China (e-mail: cswmzuo@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2018.2875346

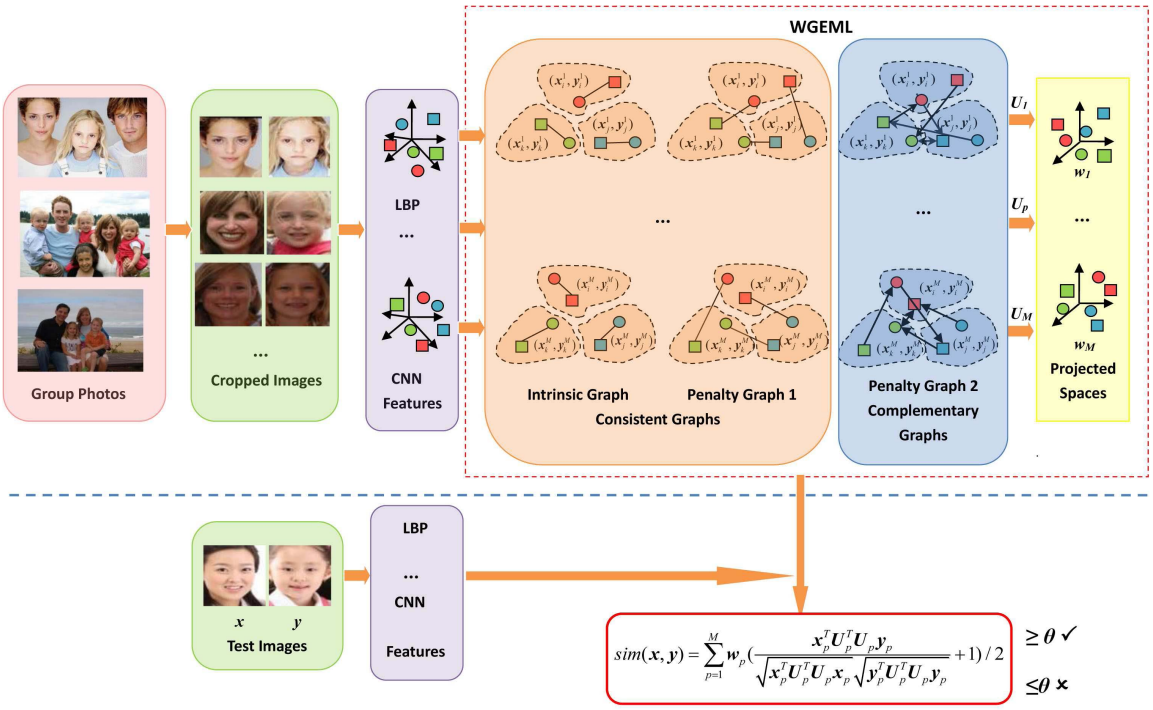


Fig. 1. Illustration of our proposed WGEML method for kinship verification, where each transformation matrix $\boldsymbol{U}_p$ and weight w_p are jointly learned for their corresponding p-th modality, $p = 1, 2, \ldots, M$. There are three pairs with mother-daughter relations, marked with red, green and blue colors. The squares and circles represent mothers and their daughters, respectively. Both the intrinsic graph and penalty graph 1 are built with the class information, which are called consistent graphs as they remain unchanged under different views. Penalty graph 2 is constructed based on the 1-nearest neighbor of the matching samples and it is a directed graph, which is called complementary graph as it may vary under different views.

above mentioned various factors on face images, significant gender differences and age gaps may exist in kinship verification. Genetic diversity brings profound difficulties to accurate description of general kinship verification patterns. According to the above discussions, it is eager to develop effective and robust metric in uncertain scenarios for improving kinship verification performance.

In this paper, we focus on learning an appropriate metric for kinship verification via facial image analysis in ambiguous environment. To make full use of both the consistency and complementarity among different types of features, we construct an intrinsic graph as well as two penalty graphs to characterize the intraclass compactness and interclass separability under each view, respectively. Fig. 1 shows the main idea of our proposed method. From Fig. 1, in the intrinsic graph, each pair with kinship relation is connected. While in the penalty graphs, pairs without kinship relation and the K-nearest neighbors of their matching samples are connected accordingly.

Our key contribution to the kinship verification problem is four-fold:

- The main challenge of kinship verification comes from the large intraclass variance and small interclass variance. To cope with this, we introduce the relative difference for modality importance evaluation, which is more objective compared with the commonly used absolute difference.
- Since each type of features can capture distinctive characteristics of images, we extract multiple visual features and fuse them together through a weighted graph embedding framework, where an intrinsic graph and two penalty graphs are constructed on each modality to promote consistency between the class information and distance metrics.
- In order to solve nonlinear kinship verification problems, we extend our method to a kernelized version by embedding each linear metric into a kernel space.
- We conduct extensive experiments to verify both the effectiveness and efficiency of our methods. Experimental results show that our methods perform favorably in kinship verification by means of facial image analysis.

The remainder of this paper is organized as follows. In Section II, we review the related work on kinship verification. The framework and solutions of our proposed WGEML are introduced in Section III, respectively. The kernelized version is developed in Section IV. In Section V, we discuss the computational complexity. The experimental results are presented in Section VI. Finally, we conclude this paper in Section VII.

II. Related Work

We briefly review two related topics: 1) kinship verification and 2) metric learning.

A. Kinship Verification

In recent years, kinship verification via face analysis has attracted great interest in computer vision community, which

is partly attributed to its lower cost and shorter time compared with biological techniques. Most of the current research mainly focuses on feature representation [20]–[22] and metric learning [11], [23], [24], which are also two essential steps of kinship verification. Typical feature descriptors include local binary pattern (LBP) [10], spatial face region descriptor (SFRD) [11] and probabilistic elastic matching (PEM) [20]. Although these hand-crafted features have shown strong capability in capturing subtle variations in uncontrollable factors like illumination and rotation, they still suffer from the limitation of handcrafted selection. Under such circumstance, several deep learning methods which aim at learning feature representation automatically from original images have been proposed recently [25]–[27]. Current representative models include auto-encoder [13], [26], deep belief network (DBN) [25], [28] and deep convolutional neural networks (DCNN) [16], [29]. Among them, DCNN has achieved unprecedented success in a variety of practical applications such as image classification [30], face recognition [16] and image retrieval [29].

In terms of distance metric, the traditional rigid Euclidean distance is not capable of mining the intrinsic underlying structure of face images. Therefore, it is necessary to learn proper metrics for enhancing performance in kinship verification.

B. Metric Learning

An enormous amount of effort has been devoted to the research of metric learning in the past decades [23], [31]–[37]. Typical algorithms include linear discriminant analysis (LDA) [36], neighborhood repulsed metric learning (NRML) [23] and marginal Fisher analysis (MFA) [37], etc. While those learned metrics can significantly improve the performance, most of them neglect the complementary information between different modalities and fail to fuse multiple features effectively.

To address these drawbacks, researchers pay attention to learning metrics with multi-modal features in recent studies [4], [12], [38]–[40]. Hu et al. [4] proposed to learn multiple metrics based on large margin concept, where both discriminative and complementary information are exploited. Yan et al. [38] utilized the maximum likelihood principle for learning discriminative multiple metrics, in which the correlation of different features is maximized. In 2017, Hu et al. [39] developed a sharable and individual multi-view metric learning approach by jointly learning multiple metrics on multi-view data. It not only retains specific property for each view but also preserves common properties for different views. In spite of the promising results obtained, these metric learning algorithms still suffer from three limitations: 1) they focus on maximizing the absolute difference of the intraclass and interclass scatterness, however, the relative difference is considered more objective for measuring differences; 2) samples that are more likely to be misclassified should be highlighted, nevertheless, most existing methods only utilize the entire intraclass/interclass variance information and consider all the samples equally in learning metrics; 3) the commonly used gradient descent strategy limits the scalability of shallow metric learning methods, while deep learning methods may encounter the small sample size problem. In this paper, by jointly learning multiple metrics on multi-feature representations, we propose an effective and efficient weighted graph embedding based metric learning method which can assign greater importance to samples that are easily-misclassified for kinship verification.

III. Metric Learning With Weighted Graph Embedding

Most existing discriminative metric learning algorithms utilize either the entire interclass/intraclass variance [31]–[33], [36] or the variance in a neighborhood [23], [37]. Although there exist several methods which can promote consistency between the distance functions and label information [34], [35], they only aim at learning metrics for a single feature space and are thus not applicable to multi-modal problems. To tackle this issue, we integrate both the available information and multiple visual features through a weighted graph embedding framework and develop a new method called WGEML.

A. Problem Statement

Suppose there are M facial representations, and let $\mathcal{S}^p = \{(\boldsymbol{x}_i^p, \boldsymbol{y}_i^p) | i = 1, 2, \ldots, N\}$ be a training positive pair set consisting of N image pairs with kinship relation in the p-th feature representation space, $\mathcal{D}^p = \{(\boldsymbol{x}_i^p, \boldsymbol{y}_j^p) | i = 1, 2, \ldots, N, j \neq i\}$ be a training negative pair set containing N image pairs without kinship relation under the p-th view, $p = 1, 2, \ldots, M$. The goal of WGEML is to seek a distance between $\boldsymbol{x}_i$ and $\boldsymbol{y}_j$

$$\begin{aligned} d^2(\boldsymbol{x}_i, \boldsymbol{y}_j) &= \sum_{p=1}^{M} w_p (\boldsymbol{x}_i^p - \boldsymbol{y}_j^p)^T \boldsymbol{A}_p (\boldsymbol{x}_i^p - \boldsymbol{y}_j^p) \\ &= \sum_{p=1}^{M} w_p d_{\boldsymbol{A}_p}^2 (\boldsymbol{x}_i^p, \boldsymbol{y}_j^p) \end{aligned} \tag{1}$$

where w_p is a weight and $\boldsymbol{A}_p$ is a $D \times D$ semidefinite positive matrix. Under each view, we will find a matrix $\boldsymbol{A}_p$ such that intraclass variance is minimized and interclass variance is maximized, meanwhile, those nearby interclass samples are pushed away. We construct two consistent graphs called intrinsic graph and penalty graph 1, and one complementary graph called penalty graph 2 to achieve these three goals, as illustrated in Fig. 1. Furthermore, we formulate the following optimization problem

$$\begin{aligned} \max_{\mathbf{A},\mathbf{w}} \ \mathcal{F} = \sum_{p=1}^{M} w_p^r [& (\frac{1}{2} (\frac{1}{NK} \sum_{i=1}^{N} \sum_{n_1=1}^{K} d_{\boldsymbol{A}_p}^2 (\boldsymbol{x}_i^p, \boldsymbol{y}_{in_1}^p) \\ & + \frac{1}{NK} \sum_{i=1}^{N} \sum_{n_2=1}^{K} d_{\boldsymbol{A}_p}^2 (\boldsymbol{x}_{in_2}^p, \boldsymbol{y}_i^p)) \\ & + \frac{1}{N} \sum_{\substack{i=1 \\ j \neq i}}^{N} d_{\boldsymbol{A}_p}^2 (\boldsymbol{x}_i^p, \boldsymbol{y}_j^p)) / \frac{1}{N} \sum_{i=1}^{N} d_{\boldsymbol{A}_p}^2 (\boldsymbol{x}_i^p, \boldsymbol{y}_i^p)] \end{aligned}$$

$$= \sum_{p=1}^{M} w_p^r [(\frac{1}{2}(\frac{1}{NK}\sum_{i=1}^{N}\sum_{n_1=1}^{K}(\boldsymbol{x}_i^p - \boldsymbol{y}_{in_1}^p)^T \boldsymbol{A}_p(\boldsymbol{x}_i^p - \boldsymbol{y}_{in_1}^p) + \frac{1}{NK}\sum_{i=1}^{N}\sum_{n_2=1}^{K}(\boldsymbol{x}_{in_2}^p - \boldsymbol{y}_i^p)^T \boldsymbol{A}_p(\boldsymbol{x}_{in_2}^p - \boldsymbol{y}_i^p)) + \frac{1}{N}\sum_{\substack{i=1 \\ j\neq i}}^{N}(\boldsymbol{x}_i^p - \boldsymbol{y}_j^p)^T \boldsymbol{A}_p(\boldsymbol{x}_i^p - \boldsymbol{y}_j^p))/ \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{x}_i^p - \boldsymbol{y}_i^p)^T \boldsymbol{A}_p(\boldsymbol{x}_i^p - \boldsymbol{y}_i^p)]$$

$$s.t. \sum_{p=1}^{M} w_p = 1, \quad w_p \geq 0, \quad p = 1, 2, \ldots, M. \tag{2}$$

where $\boldsymbol{y}_{in_1}^p$ stands for the n_1-th nearest neighbor of $\boldsymbol{y}_i^p$ and $\boldsymbol{x}_{in_2}^p$ denotes the n_2-th nearest neighbor of $\boldsymbol{x}_i^p$, accordingly. To make better use of complementary information from multiple views and avoid over-fitting, we impose w_p^r ($r > 1$) on the p-th view in the objective function. The denominator of the objective aims to pull intraclass samples $\boldsymbol{x}_i^p$ and $\boldsymbol{y}_i^p$ closer. Conversely, the numerator attempts to push interclass samples $(\boldsymbol{x}_i^p, \boldsymbol{y}_j^p)$ as well as those neighbors of their matching samples $(\boldsymbol{x}_i^p, \boldsymbol{y}_{in_1}^p)$ and $(\boldsymbol{x}_{in_2}^p, \boldsymbol{y}_i^p)$ far.

Since $\boldsymbol{A}_p$ is symmetric and positive semidefinite, we may find a nonsquare matrix $\boldsymbol{U}_p$ of size $D \times d$, with $d \ll D$, such that

$$\boldsymbol{A}_p = \boldsymbol{U}_p \boldsymbol{U}_p^T \tag{3}$$

Hence, the squared Mahalanobis distance between $\boldsymbol{x}_i^p$ and $\boldsymbol{y}_j^p$ is

$$\begin{aligned} d_{\boldsymbol{A}_p}^2(\boldsymbol{x}_i^p, \boldsymbol{y}_j^p) &= (\boldsymbol{x}_i^p - \boldsymbol{y}_j^p)^T \boldsymbol{A}_p(\boldsymbol{x}_i^p - \boldsymbol{y}_j^p) \\ &= (\boldsymbol{x}_i^p - \boldsymbol{y}_j^p)^T \boldsymbol{U}_p \boldsymbol{U}_p^T (\boldsymbol{x}_i^p - \boldsymbol{y}_j^p) \\ &= (\tilde{\boldsymbol{x}}_i^p - \tilde{\boldsymbol{y}}_j^p)^T (\tilde{\boldsymbol{x}}_i^p - \tilde{\boldsymbol{y}}_j^p) \end{aligned} \tag{4}$$

where $\tilde{\boldsymbol{x}}_i^p = \boldsymbol{U}_p^T \boldsymbol{x}_i^p \in \mathbb{R}^d$ and $\tilde{\boldsymbol{y}}_j^p = \boldsymbol{U}_p^T \boldsymbol{y}_j^p \in \mathbb{R}^d$.

Therefore, (2) can be rewritten as the following constrained optimization problem

$$\max_{\mathbf{U},\mathbf{w}} \sum_{p=1}^{M} w_p^r \frac{\mathrm{tr}[\boldsymbol{U}_p^T(\frac{1}{2}(\boldsymbol{D}_{1p} + \boldsymbol{D}_{2p}) + \boldsymbol{D}_p)\boldsymbol{U}_p]}{\mathrm{tr}[\boldsymbol{U}_p^T \boldsymbol{S}_p \boldsymbol{U}_p]}$$

$$s.t.\ \boldsymbol{U}_p^T \boldsymbol{U}_p = \boldsymbol{I}_p, \quad \sum_{p=1}^{M} w_p = 1, \quad w_p \geq 0, \quad p = 1, 2, \ldots, M. \tag{5}$$

where

$$\boldsymbol{S}_p = \frac{1}{N}\sum_{(\boldsymbol{x}_i^p, \boldsymbol{y}_i^p) \in \mathcal{S}^p} (\boldsymbol{x}_i^p - \boldsymbol{y}_i^p)(\boldsymbol{x}_i^p - \boldsymbol{y}_i^p)^T$$

$$\boldsymbol{D}_p = \frac{1}{N}\sum_{(\boldsymbol{x}_i^p, \boldsymbol{y}_j^p) \in \mathcal{D}^p} (\boldsymbol{x}_i^p - \boldsymbol{y}_j^p)(\boldsymbol{x}_i^p - \boldsymbol{y}_j^p)^T$$

Algorithm 1 Weighted Graph Embedding Based Metric Learning

Input: Training positive pair set $\{\mathcal{S}^p\}_{p=1}^M = \{(\boldsymbol{x}_i^p, \boldsymbol{y}_i^p) | i = 1, 2, \ldots, N\}_{p=1}^M$ and negative pair set $\{\mathcal{D}^p\}_{p=1}^M = \{(\boldsymbol{x}_i^p, \boldsymbol{y}_j^p) | i = 1, 2, \ldots, N, j \neq i\}_{p=1}^M$ from M views, tuning parameter r, neighborhood size K.

Output: Transformation matrices $\{\boldsymbol{U}_p\}_{p=1}^M$ and combination weights $\{w_p\}_{p=1}^M$ for multiple features.

1: **for** $p = 1$ to M **do**
2: Search K-nearest neighbors of $\boldsymbol{x}_i^p$ and $\boldsymbol{y}_i^p$ with the Euclidean distance, $i = 1, 2, \ldots, N$.
3: Construct $\boldsymbol{S}_p$,$\boldsymbol{D}_p$,$\boldsymbol{D}_{1p}$ and $\boldsymbol{D}_{2p}$ using (6).
4: Modify $\boldsymbol{S}_p$ with (10).
5: Obtain $\boldsymbol{U}_p$ by solving (9).
6: **end for**
7: Compute $\mathbf{w}$ with (13).

$$\boldsymbol{D}_{1p} = \frac{1}{NK}\sum_{\substack{(\boldsymbol{x}_i^p, \boldsymbol{y}_i^p) \in \mathcal{S}^p \\ \boldsymbol{y}_k^p \in \mathcal{N}_K(\boldsymbol{y}_i^p)}} (\boldsymbol{x}_i^p - \boldsymbol{y}_k^p)(\boldsymbol{x}_i^p - \boldsymbol{y}_k^p)^T$$

$$\boldsymbol{D}_{2p} = \frac{1}{NK}\sum_{\substack{(\boldsymbol{x}_i^p, \boldsymbol{y}_i^p) \in \mathcal{S}^p \\ \boldsymbol{x}_k^p \in \mathcal{N}_K(\boldsymbol{x}_i^p)}} (\boldsymbol{x}_k^p - \boldsymbol{y}_i^p)(\boldsymbol{x}_k^p - \boldsymbol{y}_i^p)^T. \tag{6}$$

B. Optimization

As the above optimization problem involves two kinds of variables $\mathbf{U}$ and $\mathbf{w}$, we apply an alternating method to jointly perform metric learning and dimensionality reduction.

First, we fix $\mathbf{w} = [w_1, w_2, \ldots, w_M]$ and then solve $\mathbf{U}$. Suppose $\mathbf{w}$ is known, then (5) can be transformed into a series of optimization problems as follows

$$\max_{\boldsymbol{U}_p^T \boldsymbol{U}_p = \boldsymbol{I}_p} \frac{\mathrm{tr}[\boldsymbol{U}_p^T(\frac{1}{2}(\boldsymbol{D}_{1p} + \boldsymbol{D}_{2p}) + \boldsymbol{D}_p)\boldsymbol{U}_p]}{\mathrm{tr}[\boldsymbol{U}_p^T \boldsymbol{S}_p \boldsymbol{U}_p]} \tag{7}$$

where $p = 1, 2, \ldots, M$.

The problem (7) is a typical non-convex form of a general trace ratio problem without closed form solution [41]. It can be converted to an alternative *ratio trace* problem

$$\max_{\boldsymbol{U}_p} \mathrm{tr}[(\boldsymbol{U}_p^T \boldsymbol{S}_p \boldsymbol{U}_p)^{-1}(\boldsymbol{U}_p^T(\frac{\boldsymbol{D}_{1p} + \boldsymbol{D}_{2p}}{2} + \boldsymbol{D}_p)\boldsymbol{U}_p)] \tag{8}$$

which can be solved rapidly with the following generalized eigenvalue decomposition

$$(\frac{1}{2}(\boldsymbol{D}_{1p} + \boldsymbol{D}_{2p}) + \boldsymbol{D}_p)u = \lambda \boldsymbol{S}_p u \tag{9}$$

where $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d$ are the top d largest eigenvalues, $u_1, u_2, \ldots, u_d$ are the corresponding eigenvectors. $\boldsymbol{U}_p = [u_1, u_2, \ldots, u_d]$ is the transformation matrix which projects samples from the original feature space $\mathbb{R}^D$ to a new low-dimensional space $\mathbb{R}^d$.

However, when D is larger than N, the matrix $\boldsymbol{S}_p$ becomes near-singular, which causes the eigenvalue decomposition impossible. This kind of situation may always occur, since

a variety of practical applications involve processing high-dimensional data, including bioinformatics, image retrieval and text mining [42]. To overcome this limitation, we add an identity matrix as a regularizer

$$\boldsymbol{S}_p = (1-\beta)\boldsymbol{S}_p + \beta\frac{\mathrm{tr}(\boldsymbol{S}_p)}{N}\boldsymbol{I} \tag{10}$$

where $0 \leq \beta \leq 1$ is a regularization parameter. In our experiment, β is set as 0.5 by default.

Second, we solve $\mathbf{w}$ with the obtained $\mathbf{U} = [\boldsymbol{U}_1, \boldsymbol{U}_2, \ldots, \boldsymbol{U}_M]$. We construct a Lagrange function as

$$L(\mathbf{w},\lambda) = \sum_{p=1}^{M} w_p^r \frac{\mathrm{tr}[\boldsymbol{U}_p^T(\frac{1}{2}(\boldsymbol{D}_{1p}+\boldsymbol{D}_{2p})+\boldsymbol{D}_p)\boldsymbol{U}_p]}{\mathrm{tr}[\boldsymbol{U}_p^T\boldsymbol{S}_p\boldsymbol{U}_p]} - \lambda(\sum_{p=1}^{M} w_p - 1) \tag{11}$$

Setting the derivative with respect to w_p and λ as zero, we have

$$\frac{\partial L(\mathbf{w},\lambda)}{\partial w_p} = r w_p^{r-1}\frac{\mathrm{tr}[\boldsymbol{U}_p^T(\frac{\boldsymbol{D}_{1p}+\boldsymbol{D}_{2p}}{2}+\boldsymbol{D}_p)\boldsymbol{U}_p]}{\mathrm{tr}[\boldsymbol{U}_p^T\boldsymbol{S}_p\boldsymbol{U}_p]} - \lambda = 0$$

$$\frac{\partial L(\mathbf{w},\lambda)}{\partial \lambda} = \sum_{p=1}^{M} w_p - 1 = 0 \tag{12}$$

Then, we get w_p as

$$w_p = \frac{(\mathrm{tr}[\boldsymbol{U}_p^T\boldsymbol{S}_p\boldsymbol{U}_p]/\mathrm{tr}[\boldsymbol{U}_p^T(\frac{\boldsymbol{D}_{1p}+\boldsymbol{D}_{2p}}{2}+\boldsymbol{D}_p)\boldsymbol{U}_p])^{1/(r-1)}}{\sum_{p=1}^{M}(\mathrm{tr}[\boldsymbol{U}_p^T\boldsymbol{S}_p\boldsymbol{U}_p]/\mathrm{tr}[\boldsymbol{U}_p^T(\frac{\boldsymbol{D}_{1p}+\boldsymbol{D}_{2p}}{2}+\boldsymbol{D}_p)\boldsymbol{U}_p])^{1/(r-1)}} \tag{13}$$

The basic procedure of WGEML method is shown in **Algorithm 1**.

IV. Kernelized Version

While linear metric learning methods have several advantages like simplicity and good generalization capability, the linear assumption may limit their application in capturing nonlinear data structures [43]–[45]. To overcome this deficiency, we extend WGEML to a nonlinear feature space induced by kernel functions and propose a novel method, weighted kernelized graph embedding metric learning (WKGEML).

A. Preliminaries

For convenience, we use $\boldsymbol{z}_1^p, \boldsymbol{z}_2^p, \ldots, \boldsymbol{z}_{2N-1}^p, \boldsymbol{z}_{2N}^p$ to represent $\boldsymbol{x}_1^p, \boldsymbol{y}_1^p, \ldots, \boldsymbol{x}_N^p, \boldsymbol{y}_N^p$, i.e., $\boldsymbol{x}_i^p \to \boldsymbol{z}_{2i-1}^p$ and $\boldsymbol{y}_i^p \to \boldsymbol{z}_{2i}^p$, where $i = 1, 2, \ldots, N$. Suppose $\boldsymbol{\phi}^p : \mathbb{R}^D \to \mathcal{F}$ is a nonlinear mapping which projects data under the p-th view from the original space $\mathbb{R}^D$ to a high-dimensional space $\mathcal{F}$. Let $\boldsymbol{K}^p = (\boldsymbol{\Phi}^p)^T\boldsymbol{\Phi}^p$, where $\boldsymbol{\Phi}^p = [\boldsymbol{\phi}^p(\boldsymbol{z}_1^p), \boldsymbol{\phi}^p(\boldsymbol{z}_2^p), \ldots, \boldsymbol{\phi}^p(\boldsymbol{z}_{2N-1}^p), \boldsymbol{\phi}^p(\boldsymbol{z}_{2N}^p)]$, and the kernel Gram matrix be $\boldsymbol{K}_{ij}^p = \boldsymbol{\phi}^p(\boldsymbol{z}_i^p)^T\boldsymbol{\phi}^p(\boldsymbol{z}_j^p)$. Thus, the squared Euclidean distance between $\boldsymbol{\phi}^p(\boldsymbol{z}_i^p)$ and $\boldsymbol{\phi}^p(\boldsymbol{z}_j^p)$ is

$$d^2(\boldsymbol{\phi}^p(\boldsymbol{z}_i^p), \boldsymbol{\phi}^p(\boldsymbol{z}_j^p)) = \boldsymbol{K}_{ii}^p + \boldsymbol{K}_{jj}^p - 2\boldsymbol{K}_{ij}^p \tag{14}$$

We denote

$$\begin{aligned}
\mathcal{S}_{\boldsymbol{\phi}}^p &= \{(\boldsymbol{z}_{2i-1}^p, \boldsymbol{z}_{2i}^p)|i = 1, 2, \ldots, N\},\\
\mathcal{D}_{\boldsymbol{\phi}}^p &= \{(\boldsymbol{z}_{2i-1}^p, \boldsymbol{z}_{2j}^p)|i = 1, 2, \ldots, N, j \neq i\},\\
\mathcal{D}_{\boldsymbol{\phi}}^{1p} &= \{(\boldsymbol{z}_{2i-1}^p, \boldsymbol{z}_{2k}^p)|(\boldsymbol{z}_{2i-1}^p, \boldsymbol{z}_{2i}^p) \in \mathcal{S}_{\boldsymbol{\phi}}^p, \boldsymbol{\phi}^p(\boldsymbol{z}_{2k}^p)\\
&\quad \in \mathcal{N}_K(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i}^p))\},\\
\mathcal{D}_{\boldsymbol{\phi}}^{2p} &= \{(\boldsymbol{z}_{2k-1}^p, \boldsymbol{z}_{2i}^p)|(\boldsymbol{z}_{2i-1}^p, \boldsymbol{z}_{2i}^p) \in \mathcal{S}_{\boldsymbol{\phi}}^p, \boldsymbol{\phi}^p(\boldsymbol{z}_{2k-1}^p)\\
&\quad \in \mathcal{N}_K(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p))\}.
\end{aligned}$$

Then the objective function in (5) turns into

$$\max_{\mathbf{U}^{\boldsymbol{\phi}},\mathbf{w}^{\boldsymbol{\phi}}} \sum_{p=1}^{M}(w_p^{\boldsymbol{\phi}})^r \frac{\mathrm{tr}[(\boldsymbol{U}_p^{\boldsymbol{\phi}})^T(\frac{1}{2}(\boldsymbol{D}_{1p}^{\boldsymbol{\phi}}+\boldsymbol{D}_{2p}^{\boldsymbol{\phi}})+\boldsymbol{D}_p^{\boldsymbol{\phi}})\boldsymbol{U}_p^{\boldsymbol{\phi}}]}{\mathrm{tr}[(\boldsymbol{U}_p^{\boldsymbol{\phi}})^T\boldsymbol{S}_p^{\boldsymbol{\phi}}\boldsymbol{U}_p^{\boldsymbol{\phi}}]} \tag{15}$$

where

$$\begin{aligned}
\boldsymbol{S}_p^{\boldsymbol{\phi}} &= \frac{1}{N}\sum_{\mathcal{S}_{\boldsymbol{\phi}}^p}(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2i}^p))(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2i}^p))^T\\
\boldsymbol{D}_p^{\boldsymbol{\phi}} &= \frac{1}{N}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^p}(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2j}^p))(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2j}^p))^T\\
\boldsymbol{D}_{1p}^{\boldsymbol{\phi}} &= \frac{1}{NK}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^{1p}}(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2k}^p))(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2k}^p))^T\\
\boldsymbol{D}_{2p}^{\boldsymbol{\phi}} &= \frac{1}{NK}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^{2p}}(\boldsymbol{\phi}^p(\boldsymbol{z}_{2k-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2i}^p))(\boldsymbol{\phi}^p(\boldsymbol{z}_{2k-1}^p)\\
&\quad - \boldsymbol{\phi}^p(\boldsymbol{z}_{2i}^p))^T.
\end{aligned} \tag{16}$$

B. Solution

According to the *Representer Theorem* [46], the projection matrix for the p-th view can be denoted as

$$\boldsymbol{U}_p^{\boldsymbol{\phi}} = \sum_{i=1}^{2N}\alpha_i^p\boldsymbol{\phi}^p(\boldsymbol{z}_i^p) = \boldsymbol{\Phi}^p\boldsymbol{\alpha}^p \tag{17}$$

where $\boldsymbol{\alpha}^p$ is the expansion coefficient vector of $\boldsymbol{\phi}^p(\boldsymbol{z}_i^p)$.

Therefore, we have

$$\begin{aligned}
&(\boldsymbol{U}_p^{\boldsymbol{\phi}})^T\boldsymbol{S}_p^{\boldsymbol{\phi}}\boldsymbol{U}_p^{\boldsymbol{\phi}}\\
&= (\boldsymbol{\alpha}^p)^T[\frac{1}{N}\sum_{\mathcal{S}_{\boldsymbol{\phi}}^p}(\boldsymbol{\Phi}^p)^T(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2i}^p))(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p)\\
&\quad - \boldsymbol{\phi}^p(\boldsymbol{z}_{2i}^p))^T\boldsymbol{\Phi}^p]\boldsymbol{\alpha}^p\\
&= (\boldsymbol{\alpha}^p)^T[\frac{1}{N}\sum_{\mathcal{S}_{\boldsymbol{\phi}}^p}(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2i}^p)(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2i}^p)^T]\boldsymbol{\alpha}^p
\end{aligned} \tag{18}$$

$$\begin{aligned}
&(\boldsymbol{U}_p^{\boldsymbol{\phi}})^T(\frac{1}{2}(\boldsymbol{D}_{1p}^{\boldsymbol{\phi}}+\boldsymbol{D}_{2p}^{\boldsymbol{\phi}})+\boldsymbol{D}_p^{\boldsymbol{\phi}})\boldsymbol{U}_p^{\boldsymbol{\phi}}\\
&= (\boldsymbol{\alpha}^p)^T[\frac{1}{2}(\frac{1}{NK}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^{1p}}(\boldsymbol{\Phi}^p)^T(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2k}^p))(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p)\\
&\quad - \boldsymbol{\phi}^p(\boldsymbol{z}_{2k}^p))^T\boldsymbol{\Phi}^p + \frac{1}{NK}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^{2p}}(\boldsymbol{\Phi}^p)^T(\boldsymbol{\phi}^p(\boldsymbol{z}_{2k-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2i}^p))\\
&\quad \times(\boldsymbol{\phi}^p(\boldsymbol{z}_{2k-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2i}^p))^T\boldsymbol{\Phi}^p) + \frac{1}{N}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^p}(\boldsymbol{\Phi}^p)^T(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p)\\
&\quad - \boldsymbol{\phi}^p(\boldsymbol{z}_{2j}^p))(\boldsymbol{\phi}^p(\boldsymbol{z}_{2i-1}^p) - \boldsymbol{\phi}^p(\boldsymbol{z}_{2j}^p))^T\boldsymbol{\Phi}^p]\boldsymbol{\alpha}^p
\end{aligned}$$

Algorithm 2 Weighted Kernelized Graph Embedding Based Metric Learning

Input: Training positive pair set $\{\mathcal{S}_{\boldsymbol{\phi}}^p\}_{p=1}^M = \{(z_{2i-1}^p, z_{2i}^p)|i = 1, 2, \ldots, N\}_{p=1}^M$ and negative pair set $\{\mathcal{D}_{\boldsymbol{\phi}}^p\}_{p=1}^M = \{(z_{2i-1}^p, z_{2j}^p)|i = 1, 2, \ldots, N, j \neq i\}_{p=1}^M$ from M views, kernel type and parameters σ, tuning parameter r, neighborhood size K.

Output: Expansion coefficients $\{\boldsymbol{\alpha}^p\}_{p=1}^M$ and combination weights $\{w_p^{\boldsymbol{\phi}}\}_{p=1}^M$ for multiple features.

1: **for** $p = 1$ to M **do**
2: Construct kernel matrix $\boldsymbol{K}^p$ with z_{2i}^p, z_{2i-1}^p and σ, $i = 1, 2, \ldots, N$.
3: Search K-nearest neighbors of $\boldsymbol{\phi}^p(z_{2i}^p)$ and $\boldsymbol{\phi}^p(z_{2i-1}^p)$ with (14) and then get $\mathcal{D}_{\boldsymbol{\phi}}^{1p}$, $\mathcal{D}_{\boldsymbol{\phi}}^{2p}$, $i = 1, 2, \ldots, N$.
4: Construct $\tilde{\boldsymbol{S}}_p^{\boldsymbol{\phi}}$,$\tilde{\boldsymbol{D}}_p^{\boldsymbol{\phi}}$,$\tilde{\boldsymbol{D}}_{1p}^{\boldsymbol{\phi}}$ and $\tilde{\boldsymbol{D}}_{2p}^{\boldsymbol{\phi}}$ using (20).
5: Obtain $\boldsymbol{\alpha}^p$ by solving (22).
6: **end for**
7: Compute $\mathbf{w}^{\boldsymbol{\phi}}$ with (23).

$$
\begin{aligned}
&= (\boldsymbol{\alpha}^p)^T [\frac{1}{2}(\frac{1}{NK}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^{1p}}(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2k}^p)(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2k}^p)^T \\
&\quad + \frac{1}{NK}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^{2p}}(\boldsymbol{K}_{.2k-1}^p - \boldsymbol{K}_{.2i}^p)(\boldsymbol{K}_{.2k-1}^p - \boldsymbol{K}_{.2i}^p)^T) \\
&\quad + \frac{1}{N}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^p}(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2j}^p)(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2j}^p)^T]\boldsymbol{\alpha}^p
\end{aligned} \tag{19}
$$

where $\boldsymbol{K}_{.i}^p = (\boldsymbol{\Phi}^p)^T \boldsymbol{\phi}^p(z_i^p)$ is the i-th column of $\boldsymbol{K}^p$.

Then, we can introduce matrices

$$
\begin{aligned}
\tilde{\boldsymbol{S}}_p^{\boldsymbol{\phi}} &= \frac{1}{N}\sum_{\mathcal{S}_{\boldsymbol{\phi}}^p}(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2i}^p)(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2i}^p)^T \\
\tilde{\boldsymbol{D}}_p^{\boldsymbol{\phi}} &= \frac{1}{N}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^p}(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2j}^p)(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2j}^p)^T \\
\tilde{\boldsymbol{D}}_{1p}^{\boldsymbol{\phi}} &= \frac{1}{NK}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^{1p}}(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2k}^p)(\boldsymbol{K}_{.2i-1}^p - \boldsymbol{K}_{.2k}^p)^T \\
\tilde{\boldsymbol{D}}_{2p}^{\boldsymbol{\phi}} &= \frac{1}{NK}\sum_{\mathcal{D}_{\boldsymbol{\phi}}^{2p}}(\boldsymbol{K}_{.2k-1}^p - \boldsymbol{K}_{.2i}^p)(\boldsymbol{K}_{.2k-1}^p - \boldsymbol{K}_{.2i}^p)^T
\end{aligned} \tag{20}
$$

and derive the objective function for the p-th view

$$
\max_{(\boldsymbol{\alpha}^p)^T\boldsymbol{\alpha}^p = \boldsymbol{I}_p} \frac{\mathrm{tr}[(\boldsymbol{\alpha}^p)^T(\frac{1}{2}(\tilde{\boldsymbol{D}}_{1p}^{\boldsymbol{\phi}} + \tilde{\boldsymbol{D}}_{2p}^{\boldsymbol{\phi}}) + \tilde{\boldsymbol{D}}_p^{\boldsymbol{\phi}})\boldsymbol{\alpha}^p]}{\mathrm{tr}[(\boldsymbol{\alpha}^p)^T\tilde{\boldsymbol{S}}_p^{\boldsymbol{\phi}}\boldsymbol{\alpha}^p]} \tag{21}
$$

Similar to (9), we obtain the optimal projections by solving the following generalized eigenvalue decomposition

$$
(\frac{1}{2}(\tilde{\boldsymbol{D}}_{1p}^{\boldsymbol{\phi}} + \tilde{\boldsymbol{D}}_{2p}^{\boldsymbol{\phi}}) + \tilde{\boldsymbol{D}}_p^{\boldsymbol{\phi}})\alpha = \lambda\tilde{\boldsymbol{S}}_p^{\boldsymbol{\phi}}\alpha \tag{22}
$$

$\boldsymbol{\alpha}^p = [\alpha_1, \alpha_2, \ldots, \alpha_d]$ is the optimal projection matrix to be solved, which is composed of the eigenvectors corresponding to the top d largest eigenvalues of (22).

Hence, $w_p^{\boldsymbol{\phi}}$ can be calculated as follows

$$
\begin{aligned}
&w_p^{\boldsymbol{\phi}} \\
&= \frac{(\mathrm{tr}[(\boldsymbol{\alpha}^p)^T\tilde{\boldsymbol{S}}_p^{\boldsymbol{\phi}}\boldsymbol{\alpha}^p]/\mathrm{tr}[(\boldsymbol{\alpha}^p)^T(\frac{\tilde{\boldsymbol{D}}_{1p}^{\boldsymbol{\phi}} + \tilde{\boldsymbol{D}}_{2p}^{\boldsymbol{\phi}}}{2} + \tilde{\boldsymbol{D}}_p^{\boldsymbol{\phi}})\boldsymbol{\alpha}^p])^{1/(r-1)}}{\sum_{p=1}^M(\mathrm{tr}[(\boldsymbol{\alpha}^p)^T\tilde{\boldsymbol{S}}_p^{\boldsymbol{\phi}}\boldsymbol{\alpha}^p]/\mathrm{tr}[(\boldsymbol{\alpha}^p)^T(\frac{\tilde{\boldsymbol{D}}_{1p}^{\boldsymbol{\phi}} + \tilde{\boldsymbol{D}}_{2p}^{\boldsymbol{\phi}}}{2} + \tilde{\boldsymbol{D}}_p^{\boldsymbol{\phi}})\boldsymbol{\alpha}^p])^{1/(r-1)}}
\end{aligned} \tag{23}
$$

The new coordinate of sample z_i^p is

$$
\tilde{z}_i^p = (\boldsymbol{U}_p^{\boldsymbol{\phi}})^T\boldsymbol{\phi}^p(z_i^p) = (\boldsymbol{\alpha}^p)^T(\boldsymbol{\Phi}^p)^T\boldsymbol{\phi}^p(z_i^p) = (\boldsymbol{\alpha}^p)^T\boldsymbol{K}_{.i}^p \tag{24}
$$

and the distance can be calculated as follows

$$
\begin{aligned}
d^2(z_i^p, z_j^p) &= (\tilde{z}_i^p - \tilde{z}_j^p)^T(\tilde{z}_i^p - \tilde{z}_j^p) \\
&= (\boldsymbol{K}_{.i}^p - \boldsymbol{K}_{.j}^p)^T\boldsymbol{\alpha}^p(\boldsymbol{\alpha}^p)^T(\boldsymbol{K}_{.i}^p - \boldsymbol{K}_{.j}^p)
\end{aligned} \tag{25}
$$

The WKGEML method is summarized in **Algorithm 2**.

V. Computational Complexity Analysis

We now briefly analyze the computational complexity of the WGEML and WKGEML methods, both of which involve M modalities. For WGEML, the time complexity includes M eigenvalue decompositions of $D \times D$ matrices and computation of $\mathbf{w}$. Time complexity of these two steps is $O(MD^3)$ and $O(M(NK + D^2))$, respectively. Thus, the computational complexity of WGEML is $O(MD^3 + MNK)$. Similarly, for WKGEML, its time complexity mainly comes from M eigenvalue decompositions of $2N \times 2N$ matrices and computation of $\mathbf{w}^{\boldsymbol{\phi}}$. Time complexity of these two steps is $O(MN^3)$ and $O(MN^2)$. Thus, the computational complexity of WKGEML is $O(MN^3)$.

VI. Experiments

In this section, we conduct extensive experiments to investigate the performance of our proposed WGEML and WKGEML methods on three publicly available datasets, i.e., KinFaceW-I [23], KinFaceW-II [23] and TSKinFace [5]. A brief description of these datasets is given at first, followed by the analysis and discussions of the experiments.

A. Datasets

KinFaceW-I [23] and KinFaceW-II [23] are collected from the Internet, including face images of public celebrities and their parents or children. The major difference is that each image pair with kinship relation of KinFaceW-I is obtained from different photos while that of KinFaceW-II comes from the same photo. There are four types of kinship relations in these two datasets, i.e., Father-Son (F-S), Father-Daughter (F-D), Mother-Son (M-S) and Mother-Daughter (M-D). The former contains 156, 134, 116 and 127 pairs of images with the above four kinship relations and the latter includes 250 pairs for each kinship relation.

TSKinFace [5] is a tri-subject kinship face dataset, which is also constructed from the family photos of public figures on the Internet. Different from the aforementioned datasets, it has three kinds of kinship relations, that is, Father-Mother-Son (FM-S), Father-Mother-Daughter (FM-D) and Father-Mother-Son-Daughter (FM-SD). There are 285, 274 and 228 groups

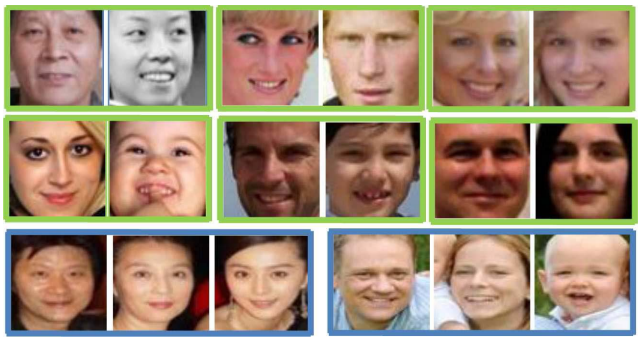


Fig. 2. Some aligned and cropped kinship pairs from three kinship datasets. From top to down are images from the KinFaceW-I, KinFaceW-II and TSKinFace datasets, accordingly. The third row shows the Father-Mother-Daughter (FM-D) and Father-Mother-Son (FM-S) relation families, respectively.

for each relation in TSKinFace, accordingly. In our experiments, we split those images into six types of relations containing 513 F-S, 502 F-D, 513 M-S, 502 M-D, 513 FM-S and 502 FM-D groups.

B. Experimental Settings

Face images in these three datasets are captured in an uncontrolled environment with no restriction in terms of age, expression, illumination and ethnicity, etc. Moreover, all the face images are aligned and cropped into 64×64 pixels. Fig. 2 presents several sample pairs.

We extract the following four types of feature descriptors for feature representation.

- Local Binary Pattern (LBP) [10]: We divide each face image into 8×8 blocks with a 8×8 grid. Then, a 59-dimensional uniform LBP descriptor is calculated for each block and a 3776-dimensional vector is obtained to represent the whole face image.
- Histogram Of Gradients (HOG) [47]: We divide each face image into 16×16 blocks using a 4×4 grid at first, then we divide image into 8×8 blocks using a 8×8 grid. A 9-dimensional feature vector is computed for each block. Finally, we concatenate all these vectors together and get a 2880-dimensional feature vector.
- SIFT: We divide each face image into 7×7 overlapping patches with a 16×16 grid. Each patch is extracted with a 128-dimensional SIFT feature vector. Thus, each face image is represented as a 6272-dimensional feature vector.
- CNN [14]: We extract the 4096-dimensional VGG-Face CNN descriptor for each face image based on the VGG-Very-Deep-16 CNN architecture.

There are two types of protocols for supervised learning on KinFaceW-I and KinFaceW-II datasets: *image-restricted* and *image-unrestricted*. In experiments, we evaluate our methods on both of these two settings. We carry out the 5-fold cross validation, where each fold owns nearly the same number of image pairs. In each fold, we exploit all pairs of face images with kin relation to form positive pairs. Meanwhile, each parent image is combined with a child image who is not his/her offspring to build negative pairs. To keep balance in learning SVM classifier, each face image appear only once in negative pairs. For kernel-based methods, we use three folds as training set, one fold as validation set and the other one as test set.

TABLE I

MEAN VERIFICATION ACCURACY (%) WITH DIFFERENT METRIC LEARNING STRATEGIES ON KINFACEW-I DATASET

Metric	Feature	F-S	F-D	M-S	M-D	avg.
GEML	LBP	73.2	71.3	71.1	73.3	72.2
	HOG	71.1	71.7	74.6	74.8	73.1
	SIFT	72.4	72.0	73.8	75.3	73.4
	CNN	77.0	69.1	78.8	78.7	75.9
CGEML	All	76.7	70.9	78.8	78.0	76.1
WGEML-com	All	78.5	73.5	79.3	80.7	78.0
WGEML	All	**78.5**	**73.9**	**80.6**	**81.9**	**78.7**

To verify the effectiveness of our methods, we take several state-of-the-art metric learning algorithms in comparison, including seven single-metric methods LDA [36], MFA [37], NRML [23], DML [38], MPDFL [48], RSBM-block-FS [5] and GMP [6], and multi-metric methods MLDA [49], MMFA [49], MNRML [23], DMML [38] and LM^{3}L [4]. For a comprehensive comparison, some deep learning methods such as CFT* [50], DDML [40], MvDML [39], SMCNN [51], Gated autoencoder [3] and DKV [52] are also included. For all the algorithms, we use PCA to project each feature representation to a 200-dimensional space and then set the reduced dimension as 100. For MFA, NRML and our proposed methods, the neighborhood size K is empirically set as 5. In our implementations, the parameter r is empirically set as 5. For DDML, we follow the same parameter setting in [40]. For WKGEML, the Gaussian kernel $\exp\{-\|\boldsymbol{x}_i - \boldsymbol{y}_j\|/\sigma^2\}$ is adopted and the scaling parameter is set as $\sigma = \gamma\sigma_0$, $\gamma =$ 16, 20, 24, 28, 32, where σ_0 is the mean square deviation of the dataset.

C. Results and Analysis

1) Comparison With Different Metric Learning Strategies: We compare our method with the following three different metric learning strategies:

- Graph Embedding based Metric Learning (GEML): we learn a single distance metric by setting $w_p = 1$ for the specific pth feature representation of (5).
- Concatenated GEML (CGEML): we concatenate multiple visual features into a long vector, and then use GEML to learn a distance metric.
- WGEML-com: we remove penalty graph 1 terms $\boldsymbol{D}_p$ $(p = 1, 2, \ldots, M)$ in (5).

Tables I-III show the mean verification accuracy of different metric learning strategies on KinFaceW-I, KinFaceW-II and TSKinFace datasets, respectively. For all kin relations, our method obtains competitive performance with other compared metric learning strategies.

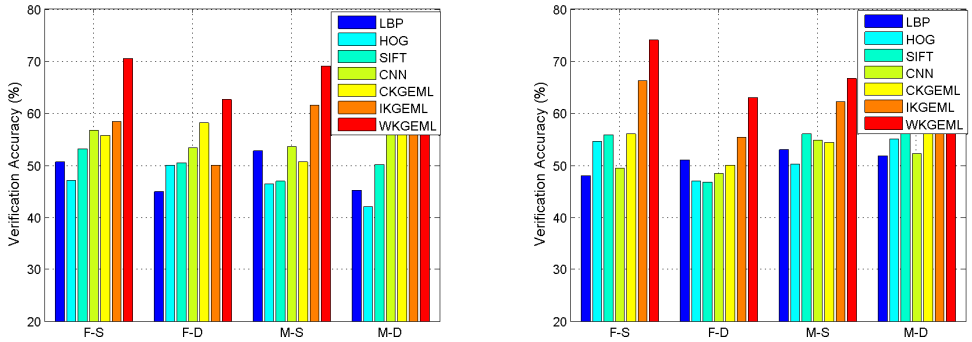


Fig. 3. Verification accuracy of different metric learning strategies with KNN classifier on KinFaceW-I and KinFaceW-II datasets. (a) KinFaceW-I. (b) KinFaceW-II.

TABLE II
MEAN VERIFICATION ACCURACY (%) WITH DIFFERENT METRIC LEARNING STRATEGIES ON KINFACEW-II DATASET

Metric	Feature	F-S	F-D	M-S	M-D	avg.
GEML	LBP	80.2	70.6	74.4	72.0	74.3
	HOG	84.8	73.2	79.4	74.2	77.9
	SIFT	82.8	74.8	80.4	76.4	78.6
	CNN	83.4	75.2	80.2	79.8	79.7
CGEML	All	83.4	75.2	79.2	80.2	79.5
WGEML-com	All	88.6	76.8	83.2	81.4	82.5
WGEML	All	**88.6**	**77.4**	**83.4**	**81.6**	**82.8**

TABLE III
MEAN VERIFICATION ACCURACY (%) WITH DIFFERENT METRIC LEARNING STRATEGIES ON TSKINFACE DATASET

Metric	Feature	F-S	F-D	M-S	M-D	FM-S	FM-D	avg.
GEML	LBP	77.2	75.5	76.7	74.8	80.9	77.8	77.1
	HOG	82.0	79.7	83.1	79.5	85.7	84.0	82.3
	SIFT	80.2	78.2	79.9	80.2	83.0	81.8	80.5
	CNN	82.0	80.4	85.5	83.3	87.1	85.9	84.0
CGEML	All	81.7	78.8	83.5	82.6	86.2	85.0	82.9
WGEML-com	All	88.3	89.8	91.4	88.7	92.4	91.5	90.4
WGEML	All	**90.3**	**89.8**	**91.4**	**90.4**	**93.5**	**93.0**	**91.4**

We compare our kernelized approach with the following metric learning strategies:

- Kernelized Graph Embedding based Metric Learning (KGEML): we learn a single distance metric by setting $w_p^{\boldsymbol{\phi}} = 1$ for the specific pth feature representation of (15).
- Concatenated KGEML (CKGEML): we concatenate multiple visual features into a long vector, and then use KGEML to learn a distance metric.
- Individual KGEML (IKGEML): we learn the distance metric for each type of features by (15), and combine these features with equal weights to compute the distance between each pair of images.

Fig. 3 presents the verification accuracy of different metric learning strategies with KNN classifier on KinFaceW-I and KinFaceW-II datasets, respectively. We observe that our WKGEML consistently outperforms all the other strategies, which is owing to both the learned proper weight as well as the metric for each feature representation. It is clear that neither CKGEML nor IKGEML can always exceed the single best feature on KinFaceW-I.

2) Comparison With Shallow Metric Learning Algorithms: Tables IV-V list the performance comparison of state-of-the-art methods as well as our proposed method with different features on KinFaceW-I and KinFaceW-II datasets under image-unrestricted and image-restricted settings, respectively. Our method under image-restricted setting does not include the complementary graph, i.e., penalty graph 2 terms $\boldsymbol{D}_{1p}, \boldsymbol{D}_{2p}, p = 1, 2, \ldots, M$ in (5). To better distinguish it from GEML/WGEML, we denote it as GEML-con/ WGEML-con. Table VI shows the verification results of different methods on TSKinFace dataset, where the best results are shown in bold. The similarity between parent-child (FM-S/D) is calculated as the mean similarity of father-child (F-S/D) and mother-child (M-S/D).

In general, our proposed WGEML method consistently outperforms all the other competing methods, which indicates that it is meaningful to exploit both the consistency and complementarity among multiple modalities in metric learning. On KinFaceW-I, WGEML outperforms MNRML and MPDFL 3.9% and 8.6% on average, respectively. On KinFaceW-II, it achieves about 2.6% lowest gains over the follower on average. Under image-restricted setting, while the number of constraints is reduced, WGEML-con still has excellent performance. It exceeds DMML 2.6% and 6.4% on KinFaceW-I and KinFaceW-II datasets, respectively. Moreover, on KinFaceW-II dataset, WGEML-con outperforms LM^{3}L more than 4%. On large datasets such as TSKinFace, our WGEML method obtains best performance on each relation. Specifically, it outperforms MNRML, DMML, RSBM-block-FS and GMP 7.6%, 7.1%, 8.4% and 2.9% in average, respectively.

Overall, multiple metrics can improve performance over the single metric, implying that the complementary information provided by multiple features is helpful for kinship verification. Among all the single visual descriptors, CNN achieves better performance compared with those hand-crafted features like LBP, HOG and SIFT, which can be attributed to the fact that it learns feature representation directly from original images rather than artificial selection.

TABLE IV
MEAN VERIFICATION ACCURACY (%) ON KINFACEW-I AND KINFACEW-II DATASETS UNDER IMAGE-UNRESTRICTED SETTING

Method	Feature	KinFaceW-I					KinFaceW-II				
		F-S	F-D	M-S	M-D	avg.	F-S	F-D	M-S	M-D	avg.
LDA [36]	LBP	68.3	65.0	69.0	66.4	67.2	71.0	64.2	67.6	66.4	67.3
	HOG	69.8	65.7	66.0	74.8	69.1	75.6	69.4	72.0	64.8	70.5
	SIFT	70.8	68.8	68.3	68.1	69.0	73.2	67.6	70.2	71.8	70.7
	CNN	72.6	66.8	72.7	74.4	71.6	77.8	66.8	72.8	74.6	73.0
MLDA [49]	All	76.6	71.2	77.7	76.4	75.5	86.6	74.4	81.0	78.8	80.2
MFA [37]	LBP	67.4	67.9	70.7	67.8	68.4	78.0	67.2	70.0	67.6	70.7
	HOG	70.5	70.0	69.9	69.7	70.1	80.2	69.6	73.6	67.8	72.8
	SIFT	69.8	71.2	69.4	70.5	70.2	78.2	70.6	73.4	69.2	72.9
	CNN	73.4	66.1	74.8	72.1	71.6	76.2	70.2	70.6	74.8	73.0
MMFA [49]	All	77.9	72.0	77.2	75.2	75.6	85.6	73.2	80.4	77.2	79.1
NRML [23]	LBP	70.2	66.4	70.3	70.1	69.3	78.0	66.2	70.0	67.2	70.4
	HOG	74.0	69.1	72.2	71.7	71.7	81.4	68.0	74.2	67.4	72.8
	SIFT	74.7	71.3	71.7	70.8	72.1	78.8	67.8	74.0	67.8	72.1
	CNN	72.4	65.7	71.8	72.6	70.6	75.6	69.2	73.6	77.2	73.9
MNRML [23]	All	76.3	69.8	77.2	75.9	74.8	82.6	68.6	76.6	73.0	75.2
MPDFL [48]	-	73.5	67.5	66.1	73.1	70.1	77.3	74.7	77.8	78.0	77.0
CFT* [50]	-	**78.8**	71.7	77.2	81.9	77.4	77.4	76.6	79.0	**83.8**	79.3
GEML	LBP	73.2	71.3	71.1	73.3	72.2	80.2	70.6	74.4	72.0	74.3
	HOG	71.1	71.7	74.6	74.8	73.1	84.8	73.2	79.4	74.2	77.9
	SIFT	72.4	72.0	73.8	75.3	73.4	82.8	74.8	80.4	76.4	78.6
	CNN	77.0	69.1	78.8	78.7	75.9	83.4	75.2	80.2	79.8	79.7
WGEML	All	78.5	**73.9**	**80.6**	**81.9**	**78.7**	**88.6**	**77.4**	**83.4**	81.6	**82.8**

TABLE V
MEAN VERIFICATION ACCURACY (%) ON KINFACEW-I AND KINFACEW-II DATASETS UNDER IMAGE-RESTRICTED SETTING

Method	Feature	KinFaceW-I					KinFaceW-II				
		F-S	F-D	M-S	M-D	avg.	F-S	F-D	M-S	M-D	avg.
DML [38]	LBP	70.2	66.4	70.3	70.1	69.3	78.0	66.2	70.0	67.2	70.4
	HOG	74.0	69.1	72.2	71.7	71.7	81.4	68.0	74.2	67.4	72.8
	SIFT	74.7	71.3	71.7	70.8	72.1	78.8	67.8	74.0	67.8	72.1
	CNN	72.4	65.7	71.8	72.6	70.6	75.6	69.2	73.6	77.2	73.9
DMML [38]	All	75.9	70.5	78.0	77.1	75.4	83.4	70.6	77.4	74.0	76.4
DDML [40]	LBP	71.5	66.8	71.6	70.5	70.1	78.4	66.0	71.2	67.8	70.9
	HOG	74.6	70.2	72.6	71.6	72.3	81.6	68.4	74.0	67.6	72.9
	SIFT	75.3	71.0	72.1	70.8	72.3	79.0	67.6	74.2	68.2	72.3
	CNN	73.3	66.1	73.5	72.6	71.4	77.0	70.4	74.4	76.8	74.7
LM^3L [4]	-	-	-	-	-	-	82.4	74.2	79.6	78.7	78.7
MvDML [39]	-	-	-	-	-	-	80.4	**79.8**	78.8	81.8	80.2
SMCNN [51]	-	75.0	**75.0**	68.7	72.2	72.7	75.0	79.0	78.0	**85.0**	79.3
Gated autoencoder [3]	-	76.4	72.5	71.9	77.3	74.5	83.9	76.7	**83.4**	84.8	82.2
GEML-con	LBP	71.2	70.9	67.3	70.5	70.0	81.2	71.6	74.6	72.6	75.0
	HOG	72.1	71.7	73.4	74.0	72.8	84.2	73.2	78.6	74.8	77.7
	SIFT	71.8	70.9	74.2	73.3	72.5	83.0	74.2	80.2	76.6	78.5
	CNN	76.7	69.8	78.3	77.5	75.6	83.4	75.6	80.6	79.8	79.9
WGEML-con	All	**78.9**	73.2	**79.4**	**80.7**	**78.0**	**88.2**	77.4	83.2	82.4	**82.8**

It is interesting to observe that the gender has a marked impact on kinship verification, that is, for almost all the cases, the verification accuracies on F/M/FM-S are relatively higher than those on F/M/FM-D under the same experimental settings. One plausible explanation is that the appearance variations of inherited females are more complicated than those of males. This is in accordance with psychological research that the kin face appearance similarity of females is less significant than that of males [53].

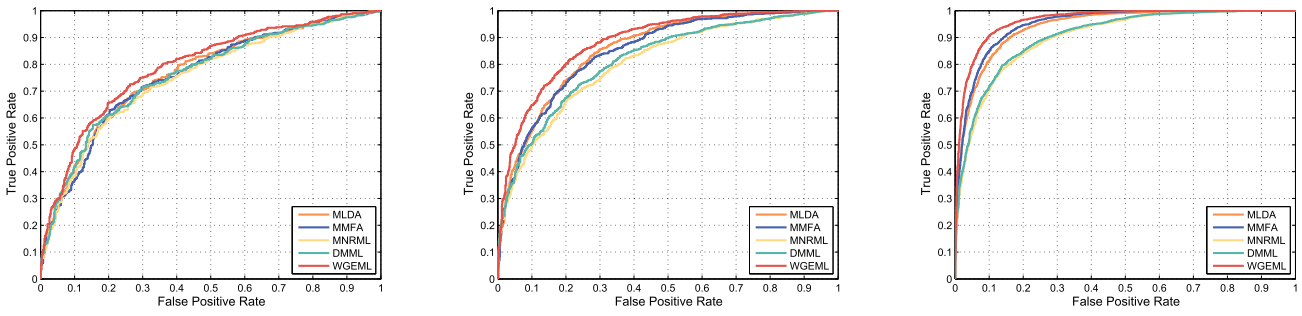


Fig. 4. ROC curves of different algorithms on kinship datasets. (a) KinFaceW-I. (b) KinFaceW-II. (c) TSKinFace.

TABLE VI
MEAN VERIFICATION ACCURACY (%) ON THE TSKINFACE DATASET

Method	Feature	F-S	F-D	M-S	M-D	FM-S	FM-D	avg.
LDA [36]	LBP	68.4	67.0	65.6	67.9	71.5	70.6	68.5
	HOG	73.9	71.3	69.6	70.2	78.8	74.9	73.1
	SIFT	70.0	67.0	69.1	70.8	73.4	73.6	70.7
	CNN	73.7	69.9	74.5	74.6	78.8	77.7	74.9
MLDA [49]	All	86.3	87.0	89.3	87.0	90.9	89.2	88.3
MFA [37]	LBP	70.7	70.0	71.6	71.5	75.6	74.7	72.3
	HOG	76.2	73.7	74.3	73.5	81.6	79.1	76.4
	SIFT	74.3	72.8	72.4	73.2	78.8	78.0	74.9
	CNN	78.7	75.5	77.4	79.4	83.1	80.0	79.0
MMFA [49]	All	89.0	87.3	90.4	88.4	91.2	90.0	89.4
NRML [23]	LBP	70.7	70.9	72.6	69.1	75.7	73.8	72.1
	HOG	75.1	73.4	74.8	69.9	78.7	76.3	74.7
	SIFT	71.8	70.8	71.4	70.0	75.6	74.4	72.4
	CNN	75.7	74.3	75.7	77.1	78.4	79.4	76.8
MNRML [23]	All	83.2	81.4	83.2	82.1	87.1	85.7	83.8
DML [38]	LBP	70.7	70.9	72.6	69.1	75.7	73.8	72.1
	HOG	75.1	73.4	74.8	69.9	78.7	76.3	74.7
	SIFT	71.8	70.8	71.4	70.0	75.6	74.4	72.4
	CNN	75.7	74.3	75.7	77.1	78.4	79.4	76.8
DMML [38]	All	83.8	81.8	84.1	82.4	87.9	86.1	84.3
DDML [40]	LBP	71.0	70.8	72.6	69.3	76.1	73.7	72.3
	HOG	75.8	73.5	75.3	70.2	78.8	77.0	75.1
	SIFT	72.6	70.9	71.6	70.3	76.3	75.2	72.8
	CNN	75.7	74.6	76.4	78.1	78.8	80.1	77.3
GMP [6]	-	88.5	87.0	87.9	87.8	90.6	89.0	88.5
RSBM-block-FS [5]	-	83.0	80.5	82.8	81.1	86.4	84.4	83.0
Gated autoencoder [3]	-	79.9	74.2	78.5	76.3	81.9	79.6	78.4
GEML	LBP	77.2	75.5	76.7	74.8	80.9	77.8	77.1
	HOG	82.0	79.7	83.1	79.5	85.7	84.0	82.3
	SIFT	80.2	78.2	79.9	80.2	83.0	81.8	80.5
	CNN	82.0	80.4	85.5	83.3	87.1	85.9	84.0
WGEML	All	**90.3**	**89.8**	**91.4**	**90.4**	**93.5**	**93.0**	**91.4**

To make intuitive comparison of our method and other metric learning algorithms, the receiver operating characteristic (ROC) curves of different multi-metric learning algorithms are presented in Fig. 4. It is clear that the ROC curves of WGEML are higher than that of other metrics.

3) Comparison With Deep Metric Learning Algorithms: In Tables IV-V, we present the experimental result of several deep learning methods including CFT*, DDML, MvDML, SMCNN and Gated autoencoder on KinFaceW-I and KinFaceW-II datasets under image-unrestricted and

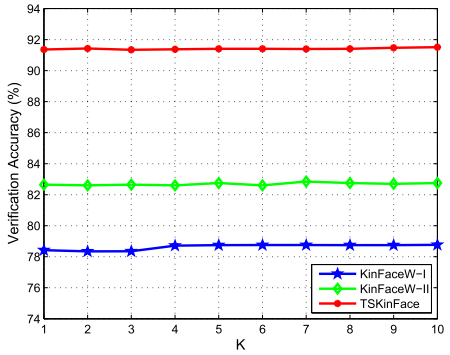


Fig. 5. Mean verification accuracy of WGEML versus different values of K on different datasets.

image-restricted settings. Compared with shallow metric learning methods such as MMFA and MNRML, CFT* improves the performance by 0.2%-4.1%, which can be attributed to its effectiveness of the coarse-to-fine transfer learning from a large image dataset. It even obtains the best performance on F-S relation of KinFaceW-I and M-D relation of KinFaceW-II with the accuracy of 78.8% and 83.8%, which exceed the follower 0.3% and 2.2%, respectively. However, on F-S relation of KinFaceW-II, it has a 11.2% lower accuracy than WGEML. For DDML, while it has slightly better performance than DMML, there is still a significant gap compared with GEML-con. The reason may be that training with deep learning model requires sufficient samples. Compared with DDML, MvDML, SMCNN and Gated autoencoder obtain relatively better performance. On KinFaceW-II dataset, these three methods achieve the highest accuracy of 79.8%, 85.0% and 83.4% on F-D, M-D and M-S relations, respectively. Nevertheless, they still have a 2.6%, 3.5% and 0.6% lower accuracy compared with WGEML-con on average. We observe that the accuracy on KinFaceW-II seems to be higher than KinFaceW-I. It is partly due to the fact that each pair of facial image in KinFaceW-II is collected from the same photo and thus sharing similar illumination conditions. Table VI lists the accuracy of DDML and Gated autoencoder on TSKinFace. It is clear that GEML significantly outperforms DDML by 5%−9% on single feature representation. On multiple feature representation, WGEML obtains 91.4% accuracy on average and has an obvious advantage in performance compared with Gated autoencoder.

4) Comparison of Different Classifiers: We evaluate performance of WGEML by employing different classifiers: NN, KNN and SVM. In experiments, we utilize LIBSVM [54] to implement SVM and choose RBF kernel as kernel function. Table VII reports the verification accuracy of different metric learning algorithms using SVM on KinFaceW-I dataset. To further analyze the performance differences, we introduce the t-test in statistics to make qualitative examinations. The significance level α is set as 0.05. The results are marked with brackets after the verification accuracy, where number "1" stands for significant difference compared with the best performance which is highlighted by bold words and "0" otherwise. On the whole, WGEML has better performance compared with other algorithms, while MNRML is inferior to other metrics,

TABLE VII
VERIFICATION ACCURACY (%) OF SVM ON KINFACEW-I

Metric	F-S	F-D	M-S	M-D
MLDA	71.5 (0)	67.1 (0)	71.1 (0)	68.2 (0)
MMFA	71.5 (0)	67.1 (0)	70.3 (0)	70.1 (0)
MNRML	63.5 (1)	56.8 (1)	60.2 (1)	59.1 (1)
DMML	71.5 (0)	67.1 (0)	70.6 (0)	72.0 (0)
DKV [52]	71.8	62.7	66.4	66.6
WGEML	**73.1**	**70.6**	**73.3**	**75.7**

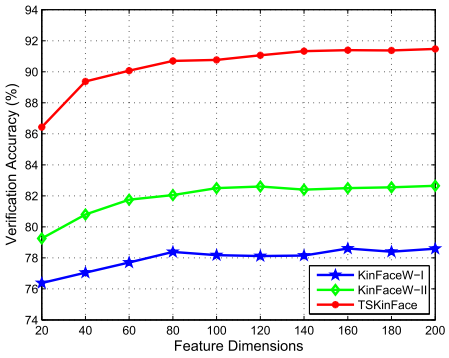


Fig. 6. Mean verification accuracy of WGEML versus different values of feature dimensions on different datasets.

which is mainly due to the fact that it projects all the features onto a common space. As for DKV, it constructs a stacked auto-encoder network with the hand-crafted LBP features, thus failing to capture the underlying kinship data structure.

Table VIII summarizes the verification accuracy when different classifiers are applied for kinship verification. For NN and KNN, the cosine similarity is utilized. As for KNN, the parameter k is set as 31 in experiments. On the whole, SVM achieves better performance compared with NN and KNN, especially on large datasets such as TSKinFace.

5) Parameter Analysis: We explore the effect of relevant parameter settings in our proposed methods. Fig. 5 shows the mean verification accuracy of WGEML versus different values of K on different datasets. We can see that WGEML has a relatively stable performance with varied neighborhood sizes, which provides convenience to choose proper parameters for performance improvement in practical applications.

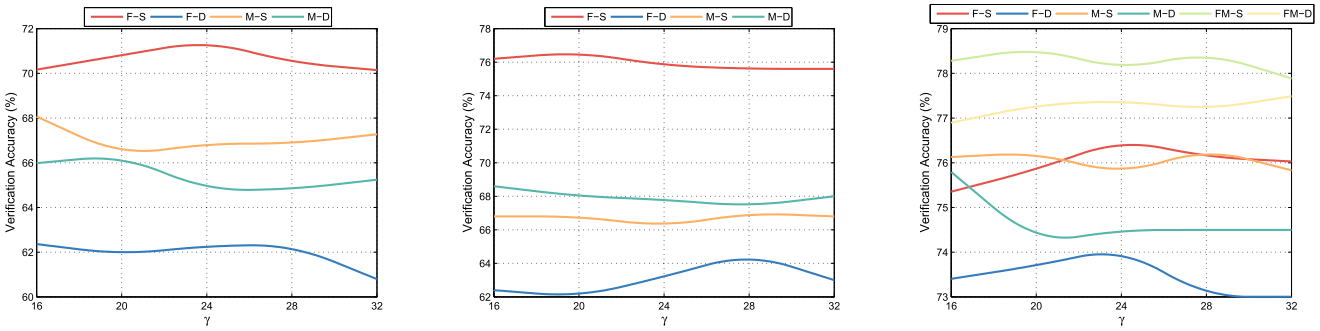
Fig. 6 plots the mean verification accuracy of WGEML versus different feature dimensions on different datasets. As the figure demonstrates, a steady verification accuracy is reached when the feature dimension exceeds 100.

Fig. 7 shows the verification accuracy of WKGEML with KNN classifier versus different values of γ on different datasets. As parameter γ varies from 16 to 32, the verification accuracy only changes within 2%, which indicates that WKGEML is not sensitive to scaling parameters in the selected interval. In addition, the curve of F-S is consistently higher than that of F-D. Therefore, we may conclude that males (or sons) share similar face appearance with their fathers than females (or daughters).

6) Computational Cost: We conduct experiments on a Windows computer (Intel i7-3770 CPU @ 3.40 GHz and 8GB RAM) with the Matlab software. Table IX lists the time

TABLE VIII
VERIFICATION ACCURACY (%) OF DIFFERENT CLASSIFIERS ON DIFFERENT DATASETS

Method	KinFaceW-I				KinFaceW-II				TSKinFace					
	F-S	F-D	M-S	M-D	F-S	F-D	M-S	M-D	F-S	F-D	M-S	M-D	FM-S	FM-D
NN	70.4	70.6	72.1	72.0	80.4	70.2	**79.0**	74.6	85.6	78.8	80.3	78.1	83.2	74.9
KNN	67.8	**70.9**	73.0	72.4	**84.8**	**73.0**	78.4	75.6	85.7	83.2	84.8	83.7	87.1	82.9
SVM	**73.1**	70.6	**73.3**	**75.7**	82.2	71.0	77.8	**77.2**	**87.8**	**86.3**	**89.7**	**88.5**	**93.3**	**91.9**



(a) (b) (c)

Fig. 7. Verification accuracy of WKGEML with KNN classifier versus different values of γ on different datasets. (a) KinFaceW-I. (b) KinFaceW-II. (c) TSKinFace.

TABLE IX
COMPUTATIONAL COST (IN SECONDS) OF DIFFERENT METRIC LEARNING ALGORITHMS ON KINFACEW-II

Metric	Training	Testing
MLDA	7.57	0.33
MMFA	7.55	0.31
MNRML	21.35	0.19
DMML	20.26	0.12
WGEML	8.17	0.13
WKGEML	14.08	0.22

cost of different multi-metric learning algorithms on KinFaceW-II dataset under such experimental setting, where the SVM classifier is used.

In training, the computational costs of MNRML and DMML are relatively larger than other methods, which is mainly due to their iterative solving strategy and time-consuming update procedure. However, their recognition time are smaller compared with other methods. As for our proposed WGEML, we observe that either its training or testing process is quite efficient in experiments.

VII. CONCLUSION AND FUTURE WORK

In this paper, we propose a weighted graph embedding metric learning (WGEML) framework for kinship verification using facial images. Unlike the current trend of metric learning which focuses on learning a common feature space or treating different types of features equally, we aim at learning multiple projections and combination weights through a joint formulation of weighted graph embedding. Meanwhile, both the intraclass compactness and interclass separability are fully captured. Furthermore, to tackle non-linear problems, a kernelized extension called WKGEML is also derived. Experimental results validate both the effectiveness and efficiency of our proposed methods.

In future, we will attempt to search a certain family that a person belongs to from a group of families. As a variety of factors including illumination, aging and poses should be considered, we will focus on extracting robust feature representation and learning effective metrics to further boost the performance.

REFERENCES

[1] M. F. Dal Martello and L. T. Maloney, "Lateralization of kin recognition signals in the human face," *J. Vis.*, vol. 10, no. 8, pp. 1–10, 2010.
[2] G. Kaminski, F. Ravary, C. Graff, and E. Gentaz, "Firstborns' disadvantage in kinship detection," *Psychol. Sci.*, vol. 21, no. 12, pp. 1746–1750, 2010.
[3] A. Dehghan, E. G. Ortiz, R. Villegas, and M. Shah, "Who do i look like? Determining parent-offspring resemblance via gated autoencoders," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1757–1764.
[4] J. Hu, J. Lu, J. Yuan, and Y.-P. Tan, "Large margin multi-metric learning for face and kinship verification in the wild," in *Proc. Asia Conf. Comput. Vis.*, 2014, pp. 252–267.
[5] X. Qin, X. Tan, and S. Chen, "Tri-subject kinship verification: Understanding the core of a family," *IEEE Trans. Multimedia*, vol. 17, no. 10, pp. 1855–1867, Oct. 2015.
[6] Z. Zhang, Y. Chen, and V. Saligrama, "Group membership prediction," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3916–3924.
[7] X. Zhou, Y. Shang, H. Yan, and G. Guo, "Ensemble similarity learning for kinship verification from facial images in the wild," *Inf. Fusion*, vol. 32, pp. 40–48, Nov. 2016.
[8] G. Guo and X. Wang, "Kinship measurement on salient facial features," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 8, pp. 2322–2325, Aug. 2012.
[9] Z. Xu, Y. Zhang, and L. Cao, "Social image analysis from a non-IID perspective," *IEEE Trans. Multimedia*, vol. 16, no. 7, pp. 1986–1998, Nov. 2014.
[10] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.
[11] Z. Cui, W. Li, D. Xu, S. Shan, and X. Chen, "Fusing robust face region descriptors via multiple metric learning for face recognition in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3554–3561.
[12] J. Lu, Y.-P. Tan, and G. Wang, "Discriminative multimanifold analysis for face recognition from a single training sample per person," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 39–51, Jan. 2013.

[13] M. Kan, S. Shan, H. Chang, and X. Chen, "Stacked progressive auto-encoders (SPAE) for face recognition across poses," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1883–1890.
[14] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. Brit. Mach. Vis. Conf.*, vol. 1, no. 3, 2015, pp. 1–12.
[15] J. Lu, V. E. Liong, X. Zhou, and J. Zhou, "Learning compact binary face descriptor for face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 2041–2056, Oct. 2015.
[16] X. Yin and X. Liu, "Multi-task convolutional neural network for pose-invariant face recognition," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 964–975, Feb. 2018.
[17] Y. Duan, J. Lu, J. Feng, and J. Zhou, "Context-aware local binary feature learning for face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 5, pp. 1139–1153, May 2018.
[18] J. Lu, V. E. Liong, and J. Zhou, "Simultaneous local binary feature learning and encoding for homogeneous and heterogeneous face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1979–1993, Aug. 2018.
[19] R. Jafri and H. R. Arabnia, "A survey of face recognition techniques," *J. Inf. Process. Syst.*, vol. 5, no. 2, pp. 41–68, 2009.
[20] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang, "Probabilistic elastic matching for pose variant face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3499–3506.
[21] X. Zhou, J. Hu, J. Lu, Y. Shang, and Y. Guan, "Kinship verification from facial images under uncontrolled conditions," in *Proc. ACM Multimedia Conf.*, 2011, pp. 953–956.
[22] X. Zhou, J. Lu, J. Hu, and Y. Shang, "Gabor-based gradient orientation pyramid for kinship verification under uncontrolled environments," in *Proc. ACM Multimedia Conf.*, 2012, pp. 725–728.
[23] J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, and J. Zhou, "Neighborhood repulsed metric learning for kinship verification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 2, pp. 331–345, Feb. 2014.
[24] S. Xia, M. Shao, J. Luo, and Y. Fu, "Understanding kin relationships in a photo," *IEEE Trans. Multimedia*, vol. 14, no. 4, pp. 1046–1056, Aug. 2012.
[25] G. B. Huang, H. Lee, and E. Learned-Miller, "Learning hierarchical representations for face verification with convolutional deep belief networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2518–2525.
[26] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng, "Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 3361–3368.
[27] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler, "Convolutional learning of spatio-temporal features," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 140–153.
[28] N. Kohli, M. Vatsa, R. Singh, A. Noore, and A. Majumdar, "Hierarchical representation learning for kinship verification," *IEEE Trans. Image Process.*, vol. 26, no. 1, pp. 289–302, Jan. 2017.
[29] X. Lu, Y. Chen, and X. Li, "Hierarchical recurrent neural hashing for image retrieval with hierarchical convolutional features," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 106–120, Jan. 2018.
[30] X. Cao, F. Zhou, L. Xu, D. Meng, Z. Xu, and J. Paisley, "Hyperspectral image classification with Markov random fields and a convolutional neural network," *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2354–2367, May 2018.
[31] P. H. Zadeh, R. Hosseini, and S. Sra, "Geometric mean metric learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 2464–2471.
[32] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 209–216.
[33] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *J. Mach. Learn. Res.*, vol. 10, pp. 207–244, Feb. 2009.
[34] S. C. H. Hoi, W. Liu, and S.-F. Chang, "Semi-supervised distance metric learning for collaborative image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–7.
[35] F. Wang, "Semisupervised metric learning by maximizing constraint margin," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 4, pp. 931–939, Aug. 2011.
[36] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, 1997.
[37] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.
[38] H. Yan, J. Lu, W. Deng, and X. Zhou, "Discriminative multimetric learning for kinship verification," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 7, pp. 1169–1178, Jul. 2014.
[39] J. Hu, J. Lu, and Y.-P. Tan, "Sharable and individual multi-view metric learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 9, pp. 2281–2288, Sep. 2018.
[40] J. Lu, J. Hu, and Y.-P. Tan, "Discriminative deep metric learning for face and kinship verification," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4269–4282, Sep. 2017.
[41] Y. Jia, F. Nie, and C. Zhang, "Trace ratio problem revisited," *IEEE Trans. Neural Netw.*, vol. 20, no. 4, pp. 729–735, Apr. 2009.
[42] J. Wang, *Geometric Structure of High-Dimensional Data and Dimensionality Reduction*. Berlin, Germany: Springer, 2011.
[43] S. Herath, M. Harandi, and F. Porikli, "Learning an invariant Hilbert space for domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 3956–3965.
[44] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. Harandi, "Kernel methods on the Riemannian manifold of symmetric positive definite matrices," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 73–80.
[45] K. Q. Weinberger and L. K. Saul, "An introduction to nonlinear dimensionality reduction by maximum variance unfolding," in *Proc. AAAI*, vol. 6, 2006, pp. 1683–1686.
[46] B. Schölkopf and A. J. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2002.
[47] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jun. 2005, pp. 886–893.
[48] H. Yan, J. Lu, and X. Zhou, "Prototype-based discriminative feature learning for kinship verification," *IEEE Trans. Cybern.*, vol. 45, no. 11, pp. 2535–2545, Nov. 2015.
[49] A. Sharma, A. Kumar, H. Daume, and D. W. Jacobs, "Generalized multiview analysis: A discriminative latent space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2160–2167.
[50] Q. Duan, L. Zhang, and W. Zuo, "From face recognition to kinship verification: An adaptation approach," in *Proc. IEEE Conf. Comput. Vis.*, Oct. 2017, pp. 1590–1598.
[51] L. Li, X. Feng, X. Wu, Z. Xia, and A. Hadid, "Kinship verification from faces via similarity metric based convolutional neural network," in *Proc. Int. Conf. Image Anal. Recognit.*, 2016, pp. 539–548.
[52] M. Wang *et al.*, "Deep kinship verification," in *Proc. IEEE Int. Workshop Multimedia Signal Process.*, 2015, pp. 1–6.
[53] S. M. Platek *et al.*, "Reactions to children's faces: Males are more affected by resemblance than females are, and so are their brains," *Evol. Human Behav.*, vol. 25, no. 6, pp. 394–405, 2004.
[54] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, 2011.
[55] X. Qin, "Research on kinship verification based on Web images," Ph.D. dissertation, Nanjing Univ. Aeronaut. Astronaut., Nanjing, China, 2016.

Jianqing Liang received the B.E. degree from the School of Computer and Information Technology, Shanxi University, Taiyuan, Shanxi, China, in 2013. She is currently pursuing the Ph.D. degree with the College of Computer Science and Technology, Tianjin University. Her current research interests include metric learning, semi-supervised learning, and machine learning.

Qinghua Hu received the B.E., M.E., and the Ph.D. degrees from the Harbin Institute of Technology, Harbin, China, in 1999, 2002, and 2008, respectively. He was with the Harbin Institute of Technology as an Assistant Professor and Associate Professor from 2006 to 2011 and a Post-Doctoral Fellow with The Hong Kong Polytechnic University. He is currently a Full Professor with Tianjin University. His research interests include intelligent modeling, data mining, and knowledge discovery for classification and regression. He has published more than 100 journal and conference papers in the areas of pattern recognition, machine learning, and data mining. He is the PC Co-Chair of RSCTC 2010, CRSSC 2012, and ICMLC 2014, and he serves as a referee for a great number of journals and conferences.

Chuangyin Dang (SM'03) received the B.S. degree in computational mathematics from Shanxi University, China, in 1983, the M.S. degree in applied mathematics from Xidian University, China, in 1986, and the Ph.D. degree in operations' research/economics from the University of Tilburg, The Netherlands, in 1991. He is currently a Professor at the City University of Hong Kong. He is best known for the development of the D1-triangulation of the Euclidean space and the simplicial method for integer programming. His current research interests include computational intelligence, optimization theory and techniques, and applied general equilibrium modeling and computation. He is a member of ES, INFORMS, and MPS.

Wangmeng Zuo received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2007. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology. His current research interests include image enhancement and restoration, object detection, visual tracking, and image classification. He has published over 70 papers in top-tier academic journals and conferences. He has served as a Tutorial Organizer in ECCV 2016, an Associate Editor of the *IET Biometrics* and the *Journal of Electronic Imaging*, and the Guest Editor of *Neurocomputing*, *Pattern Recognition*, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.