Journal of Universal Computer Science, vol. 29, no. 4 (2023), 326-348
submitted: 18/6/2022, accepted: 29/12/2022, appeared: 28/4/2023 CC BY-ND 4.0

Feature Fusion and NRML Metric Learning for Facial Kinship Verification

Fahimeh Ramazankhani
(Department of Computer Engineering, Faculty of Engineering, Yazd University, Yazd, Iran
https://orcid.org/0000-0002-9029-2494, framazankhani@stu.yazd.ac.ir)

Mahdi Yazdian-Dehkordi*
(Department of Computer Engineering, Faculty of Engineering, Yazd University, Yazd, Iran
https://orcid.org/0000-0002-4161-8638, yazdian@yazd.ac.ir)

Mehdi Rezaeian
(Department of Computer Engineering, Faculty of Engineering, Yazd University, Yazd, Iran
https://orcid.org/0000-0003-1961-8495, mrezaeian@yazd.ac.ir)

Abstract: Features extracted from facial images are used in various fields such as kinship verification. The kinship verification system determines the kin or non-kin relation between a pair of facial images by analysing their facial features. In this research, different texture and color features have been used along with the metric learning method, to verify the kinship for the four kinship relations of father-son, father-daughter, mother-son and mother-daughter. First, by fusing effective features, NRML metric learning used to generate the discriminative feature vector, then SVM classifier used to verify to kinship relations. To measure the accuracy of the proposed method, KinFaceW-I and KinFaceW-II databases have been used. The results of the evaluations show that the feature fusion and NRML metric learning methods have been able to improve the performance of the kinship verification system. In addition to the proposed approach, the effect of feature extraction from the image blocks or the whole image is investigated and the results are presented. The results indicate that feature extraction in block form, can be effective in improving the final accuracy of kinship verification.

Keywords: kinship verification, feature fusion, metric learning, NRML metric learning, facial analysis
Categories: I.4.7, I.5, I.6.4, I.2.6
DOI: 10.3897/jucs.89254

1 Introduction

Nowadays, different practical applications benefit from intelligent analysis of personal information such as gender, age, nationality, etc. In this context, DNA is one of the most accurate ways to access this information. However, DNA is not always accessible [Laiadi et al., 2019]. The facial characteristics of a person provide different unique features, including biological features. These features can be used in information extraction like gender [Swaminathan et al., 2020, Kale et al., 2021, Alghaili et al., 2020], age [Agbo-Ajala et al., 2021, Dagher et al., 2021], identity [Ratnaparkhi et al., 2021, Sudhakar et al., 2021], Facial behaviours [Kim, 2021], etc. Facial kinship verification is a noticeable research field these years. Facial kinship verification intends to verify kin or non-kin relation between a pair of face images. The Kinship verification system can be used in various applications such as forensics [Laiadi et al., 2019], find missing persons [Wu et al., 2018], make a family album, analyze facial images shared in social media [Lopez et al., 2018], and refugee crisis [Robinson et al., 2018]. The facial images technologies have also some ethical consequences such as abuse and violation of citizens' rights [Zhu et al., 2021], illegal surveillance and discrimination using facial images against people, especially the minority of society [Smith et al., 2022], violation of privacy, self-safty and self-regard [Royakkers et al., 2018]. Some mechanisms should also be considered to minimize these threats and avoid harmful consequences.

In recent years, metric learning has attracted a lot of attention in different applications [Liu et al., 2022, Tang et al., 2022, Guo et al., 2022]. Besides, color and texture features achieve effective results on kinship verification [Wu et al., 2021, Van et al., 2019, Lu et al., 2012]. Accordingly, the most focus of our research in on metric learning in conjunction with color and texture features for kinship verification.

Generally, eleven kin relationships can be defined in three levels: same generation, first-generation, and second-generation [Robinson et al., 2018]. Same generation kin relations are brother-brother, sister-sister, brother-sister. First-generation kin relations are father-son, father-daughter, mother-son, and mother-daughter. Finally, second-generation kin relations are grandfather-grandson, grandfather-granddaughter, grandmother-grandson, and grandmother-granddaughter.

The most critical challenge in kinship verification is collecting appropriate training images. Gathered images from family members have different conditions that will affect the final accuracy of the system. Some challengeable conditions are image resolution change, illumination change, blur image, complex or crowded background [Wu et al., 2019], parents and children age gap, different races, and gender difference [Wang et al., 2019]. Here, we need to extract discriminative features from the images to develop an efficient kinship verification system. General and shallow features from images can extract by texture and color features [Ramazankhani et al., 2021]. One of the common methods in kinship verification is using these two features separately or fuse them. In 2018, Wang et al. [Wang et al., 2019] proposed a kinship verification method by considering the age gap between parent and child. For this, a Generative Adversarial Networks (GANS) is used to generate rejuvenated images of the old parents' images. The authors extracted the texture features, including Local Binary Patterns (LBP) and Scale Invariant Feature Transformation (SIFT) and deep features using RENSET network. Then, in order to find a distance matrix from pair images, they utilize cosine similarity as well as several metric learning methods, including Locality Preserving Projections (LPP), Neighborhood Preserving Embedding (NPE), Large Margin Nearest Neighbor (LMNN), and Sparse Discriminative Metric Loss (SDM-Loss) algorithms. In 2015, Jiwen Lu et al. [Lu et al., 2015] used Histograms Of Oriented Gradients (HOG) and LBP features and Principal Component Analysis (PCA) algorithm to reduce feature vector dimensions. Neighborhood Repulsed Metric Learning (NRML) is used to obtain the distance metric between intraclass samples (with kinship relations) and interclass samples (without kinship relations). Goyal et al. [Goyal et al., 2020] in 2020 introduced a template matching method. In this method, the parent/child facial image components (eyes, nose, and lips) are detected by the viola joes pattern. Then, using morphological functions, accurate images of facial components are extracted. Next, the Normalized cross-correlation method (NCC) algorithm is used to match the pattern between each component of the parent and child. Finally, the components of the face are merged and after calculating the similarity score for each component, the maximum amount of similarity is selected. In 2021, Wu et al. [Wu et al., 2021] introduced an automated kinship verification system. In this system, the parent and child facial images are first divided into blocks of local facial features. Each block is then converted to feature vectors using the SIFT descriptor. After combining the parent and child facial image feature vectors, their relationship is evaluated by component-based metric learning.

Color features are another method for feature extraction for kinship verification. In 2018, Xiaoting Wu et al. [Wu et al., 2018] suggested a method that analyzes images in HSV color space. In this method, parent and child RGB input images are converted to HSV color space. Then, for each channel, Binarized Statistical Image Features (BSIF) have been produced. Two feature vectors extracted from parent and child images are fused by the cosine similarity criterion. Finally, to verify the kinship relation, the extreme machine learning model has been learned. In 2019, Van et al. [Van et al., 2019] proposed a kinship verification method using LBP feature in YUV and bwrgb color spaces. Each image is divided into non-overlapping blocks at different levels. For every block, the LBP feature has been extracted. Finally, a Support Vector Machine (SVM) is used to confirm or deny the kinship relation. Laiadi et al. [Laiadi et al., 2019] in 2019 studied the performance of several color space information in facial kinship verification. In this study, input RGB images are converted to YcbCr, LUV, Lab, HSV, HSl, and YUV color spaces. Afterward, local features like BSIF, Local Phase Quantization (LPQ), and Co-Occurrence Of Adjacent Local Binary Patterns (COALBP) are extracted for each channel. For each color space, cosine similarity has been obtained between the feature vectors of parent and child. Then, a fusion score has been computed using a Logistic regression method. Finally, the fusion score will be compared with a threshold to determine the positive or negative kinship relation. In 2020, Ravi Kumar et al. [Ravi Kumar et al., 2020], introduced a descriptor for kinship verification by the inner pixel similarity and the full binary tree. In this method, which is inspired by the LBP descriptor, the input image is divided into its R, G, and B color channels. Then, a 3-by-3 block navigates each of the color channels. In this navigation, by comparing the central pixel and its 8 neighboring pixels, the input image becomes a binary pattern. By comparing the central pixels and the neighboring pixels, their path of similarity is traced and weighed. Similar patterns in different channels are combined and a descriptor is created. Finally, the NRML metric learning is used to verify the kinship.

In recent years, with the availability of high-power hardware resources, deep neural networks have also been used for facial kinship verification.

In 2018, Lopez et al. [Lopez et al., 2018] provided a kinship verification method by extracting deep and texture features (LBP) from images. Deep features have been extracted from video frames of faces using a convolutional neural network. Laiadi and his colleagues [Laiadi et al., 2019] in 2019 provided a kinship verification method for second-generation relationships (grandfather/grandmother and grandchild). In this study, deep features have been extracted using VGG[1] -Face neural network. In 2019 Nandy et al. [Nandy et al., 2019] introduced Deep Siamese Convolutional Neural Network to verify kinship by extracting deep features. The SqueezeNet network has been used to enhance the quality of the extracted features. For each pair of parent and child, feature vectors are generated using the SqueezeNet network. By applying the

[1] Visual Geometry Group

cosine similarity criteria and Euclidean distance, the two feature vectors are merged. Finally, by sigmoid activation function with output between 0 and 1, kinship or non-kinship relation is detected. Bisogni et al. [Bisogni et al., 2022], introduced an automated kinship verification system using a deep learning network. In this research, two parallel structures of VGG-Face16 are used in the Siamese neural network. Also, due to the lack of enough facial images, different transformations such as vertical or horizontal rotation, variety of brightness, horizontal flip, vertical flip, etc. have been used. Laiadi et al. [Laiadi et al., 2020] employed several pre-trained models, including VGG-F, VGG-M, VGG-S, and VGG-Face, to extract deep facial feature for kinship verification. In 2021, Yan et al. [Yan et al., 2021] used a deep-relational network in which parent and child facial images were used as the input of two networks with shared weights. In this network, images are converted to features at three different scales depending on the kernel size. Then, by using obtained local features in the deep-relational network, kinship or non-kinship relation would verify. Wang et al. [Wang et al., 2020] proposed a two-step kinship verification system. In the first step, a set of negative examples is generated and scored using a pre-trained network. In the second step, discriminating negative samples are selected to train a kinship verification network. Dornaika et al. [Dornaika et al., 2019] examined the combination of deep features and the integration of kinship classifiers. At first, the deep features of parent and child images are extracted by two VGG-Face and VGG-F descriptors. The four feature vectors generated by Fisher score are quantified, and the features with the most correlation are selected. The four final vectors are evaluated by SVM classifiers, and their results are merged together. The major challenge in deep learning methods is to collect a large number of data and label them, which needs a lot of time and money. In addition, training these methods and evaluating and testing them requires powerful hardware and processing resources, which carry a high cost, and provision of these costs leads to limited use. As mentioned in previous researches, one of the main challenges in developing an efficient kinship verification system is extracting features from images.

In this research, we propose two NRML metric learning-based strategies, and a Siamese convolutional neural Network for kinship verification problem. The NRML method calculates a metric that aims at discriminating intraclass (with kinship relations) and interclass (without kinship relations) samples. It utilizes a thresholding approach to perform the final classification. Here, the performance of different features and their fusions have been analyzed in combination with the NRML method. Besides, an SVM-based approach trained over NRML similarity output of samples is also proposed. In addition, a Siamese convolutional neural network has been designed to verify the kinship relations based on deep features. To study the efficiency of the methods, they have been evaluated in different conditions on two well-known kinship verification datasets. In the rest of the paper, texture features and metric learning are introduced in section 2. Descriptions of the proposed methods are given in Section 3. The proposed methods are evaluated and reviewed in section 4. Finally, conclusions are given in section 5.

2 BACKGROUND

Facial kinship verification researches show that feature selection and feature fusion are effective in verification accuracy. This section provides a brief description of some of the texture features, feature descriptors, and metric learning methods used in this research.

A. Texture Features

In this paper, various texture features have been used to verify, identify and analyze human facial images, including LBP, HOG, BSIF, PML, RCM, and GRCM texture features.

Local Binary Patterns (LBP)

LBP descriptor is a powerful texture feature extracting operator from images. All image pixels will be labeled by moving a window on the image, and each central pixel is compared to its 8-neighboring pixels. The neighboring pixels that are smaller than the central pixel are denoted by 0, and the others are denoted by 1. Then, the resulted 8-digit binary number is converted to decimal. Finally, the LPB descriptor is defined as the histogram of values over a cell (8×8 blocks) [Moujahid et al., 2019].

Histogram of Oriented Gradients (HOG)

Histogram of oriented gradients descriptor counts the number of oriented gradients in the image area and returns the histogram [Moujahid et al., 2019].

Binarized Statistical Image Features (BSIF)

Binarized statistical image feature is a texture descriptor that uses a binary code like LBP descriptor. The BSIF method convolves input images with linear filters and returns the filter response in binary. The code is generated for each pixel of the image by binarizing the filter response using a zero threshold. The filters have been learned over some natural images to maximize the filters' statistical independence [Kannala et al., 2012].

Quaternionic Local Ranking Binary Pattern (QLRBP)

Quaternion is a four-dimension complex number which can be defined as follows [Lan et al., 2016]:

$$\dot{q} = a + ib + jc + kd$$

where $\dot{q}$ has one real part and three imaginary parts. Also, a, b, c, and d are real numbers and i, j, and k are complex operators. In 2016 Lan et al. [Lan et al., 2016] proposed a local color image descriptor called quaternionic local ranking binary pattern (QLRBP). Unlike the traditional descriptors that extract features from each color channel separately, QLRBP descriptor uses quaternionic representation of color images to encode all color channels [Lan et al., 2016]. To show a color pixel by its quaternionic representation, the real part is set to zero and the imaginary parts are defined by:

$$\dot{q} = ir + jg + kb$$

where r, g, and b denote the red, green, and blue channels.

QLRBP handles the color channels directly in quaternionic domain and includes their relations simultaneously. It ranks each two color pixels using a reference quaternion and generates a local descriptor by performing a local binary coding on the pixels. After coding the pixels, QLRBP processes the coding images with overlapping blocks and calculates the normalized histogram to form the final QLRBP feature vector.

Pyramid Multi-Level (PML) face representation

In Pyramid Multi-Level (PML), facial images are presented in a pyramid at different scales. Each image is processed in a multi-block representation. The feature vectors are extracted from the blocks at each level. Finally, the feature vectors are concatenated to form the final feature vector. [Fig. 1] shows the PML structure for a three-level pyramid [Moujahid et al., 2019].

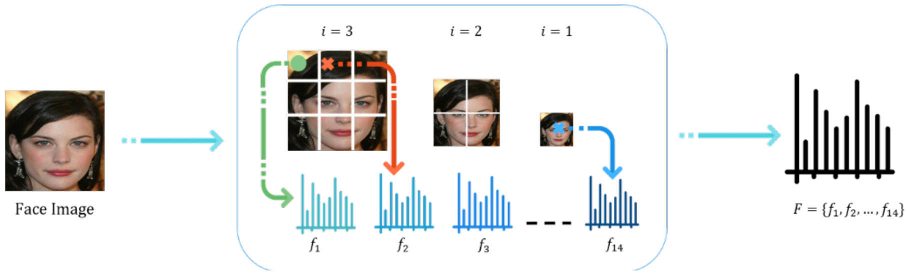


Figure 1: Multilevel Pyramid Descriptor (PML) for a three-level pyramid

Region Covariance Matrix (RCM)

The RCM feature descriptor was introduced in 2006 by Tuzel [Tuzel et al., 2006]. RCM is a covariance matrix of several statistical vectors calculated from image regions. This descriptor is inherently a way to fuse several features of an image area into a matrix. For the image I, the function $\emptyset$ extracts the d-dimensional feature vector z_i from each pixel (x,y) located in region R:

$$z_i = \emptyset(I, x, y) \qquad z_i \in \mathbb{R}^d \tag{1}$$

For region R with n pixel and $\{z_i\}_{i=1..n}$ feature vectors, the RCM is calculated as equation 2:

$$C_R = \frac{1}{n-1}\sum_{i=1}^{n}(z_i - \mu_R)(z_i - \mu_R)^T \tag{2}$$

where μ_R is the mean of z_i.

Tuzel et al. [Tuzel et al., 2006] proposed the mapping function with pixel locations, pixel values in RGB color space, and norms of first and second-order derivatives with respect to x and y. Equation 3 shows this feature vector.

$$\emptyset(I, x, y) = [\, x \quad y \quad R(x,y) \quad G(x,y) \quad B(x,y) \quad |I_x| \quad |I_y| \quad |I_{xx}| \quad |I_{yy}| \,] \tag{3}$$

PML-COV

In 2018 Moujahid et al. [Moujahid et al., 2019] proposed PML-COV descriptor based on Pyramid Multi-Level image representation. This descriptor extracts RCM features for each block at the pyramid level and concatenates them. As shown in equation 4, the mapping function considered in this study consists of pixel locations (x ,y), pixel values in RGB and HSV[2] color spaces, the norm of the first and second-order derivatives of the intensities with respect to x and y, LBP feature vector in three-mode (uniform, rotation invariant and both of them) and QLRBP descriptor [Moujahid et al., 2019].

[2] Hue, Saturation and Value

$$\emptyset(\mathrm{I},\mathrm{x},\mathrm{y}) = \begin{bmatrix} x \quad y \quad R(x,y) \quad G(x,y) \quad B(x,y) \quad H(x,y) \quad S(x,y) \quad V(x,y) \\ |\mathrm{I_x}| \quad |\mathrm{I_y}| \quad |\mathrm{I_{xy}}| \quad |\mathrm{I_{yx}}| \quad |\mathrm{I_{xx}}| \quad |\mathrm{I_{yy}}| \\ LBP_{u2} \quad LBP_{ri} \quad LBP_{riu2} \quad QLRBP_{p1} \quad QLRBP_{p2} \quad QLRBP_{P3} \end{bmatrix} \tag{4}$$

Gabor-wavelet-based Region Covariance Matrix (GRCM)

In 2008, Pang et al. [Pang et al., 2008] used Gabor features along with pixel locations to increase the performance of the RCM descriptor. A 2-D Gabor wavelet transformation is applied on the image. The real parts of the Gabor kernels are shown in [Fig. 2]. These kernels are expected to provide more information than the first and second-order derivatives of the gradient.

The Gabor features are extracted by convolving the Gabor kernels φ_{uv} with image I:

$$g_{uv}(x,y) = |\mathrm{I}(x,y) * \varphi_{uv}(x,y)| \tag{5}$$

Here, the location of the pixels, their intensities, and the Gabor features [Pang et al., 2008] form the final mapping function to extract the GRCM descriptors, i.e.:

$$\emptyset(\mathrm{I},\mathrm{x},\mathrm{y}) = [\, x \quad y \quad I(x,y) \quad g_{00}(x,y) \quad g_{01}(x,y) \ \dots \ g_{74}(x,y)\,] \tag{6}$$

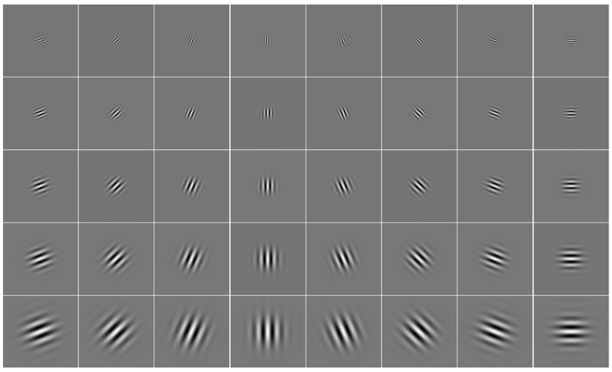


Figure 2: Real part of Gabor kernels with eight orientations (u∈{0,...,7}) and five scales (v∈{0,...,4}).

B. Metric Learning

Most machine learning methods use distance metric to identify the distance between samples. Traditionally, standard distance metrics such as Euclidean distance, Cosine similarity and Manhattan distance can be used when there is no prior information about data. However, in practice these metric might be incompatible with real-world data. In recent years, metric learning has been proposed to automatically learn a distance function for a particular data.

Jiwen Lu et al. [Lu et al., 2012] proposed Neighborhood Repulsed Metric Learning (NRML) for kinship verification. This method aims at learning a distance metric that projects the facial images with kinship relations as close as possible and pulls those without kinship relations as far as possible. Let $S = \{(x_i, y_i) | i = 1,2,,\dots,N\}$ be the training set that (x_i, y_i) shows a (parent, children) relationship. The NRML method finds the metric d that x_i becomes closer to y_j (when i=j) , and x_i becomes farther from y_j (when i≠j):

$$d(x_i, y_j) = \sqrt{(x_i - y_j)^T A(x_i - y_j)} \tag{7}$$

In this equation, A is a symmetric and positive semidefinite matrix defined as $A = W^T W$. To find the distance, the NRML method try to maximize the following objective function:

$$J(A) = \frac{1}{Nk}\sum_{i=1}^{N}\sum_{t_1=1}^{k} d^2(x_i, y_{it_1}) + \frac{1}{Nk}\sum_{i=1}^{N}\sum_{t_2=1}^{k} d^2(x_{it_2}, y_i) - \frac{1}{N}\sum_{i=1}^{N} d^2(x_i, y_i) \tag{8}$$

where y_{it_1} is the t_1 th nearest neighbor to y_j, and x_{it_2} is the t_2 th nearest neighbor to x_i, respectively.

3 PROPOSED METHOD

Metric learning is learning a distance function from samples to model similarity and dissimilarity between objects. In this research, the NRML metric learning has been studied along with various texture and color features as well as their fusion. Also, an SVM classifier trained over the NRML similarity output of samples has been proposed to verify the kinship relations automatically. Finally, a Siamese neural network design and its performance in kinship verification have been investigated. The proposed approaches have been described in the following sections in more detail.

- Feature fusion and metric learning

As mentioned earlier, metric learning algorithms have been introduced to adapt standard distance metrics to the data [Bellet et al., 2013]. In the first approach, the performance of various features and their combinations have been investigated with the NRML method to find the best feature vector. [Fig. 3] shows the overview of the method.

In the training stage ([Fig. 3-A]) the set of features [$f_1, f_2, f_3, \dots, f_n$] is extracted for each parent/child pairs. In the [Fig. 3], the parent and child feature vectors are shown by F_P and F_C, respectively. Then, min-max normalization is used to normalize the feature vectors:

$$F = \frac{F - \min(F)}{\max(F) - \min(F)} \tag{9}$$

Afterward, the W distance metric is calculated using the NRML metric learning method described in the background section.

In the test stage ([Fig. 3-B]), the extracted features F_P *and* F_C are mapped into a new feature space using the learned W distance metric:

$$F'_P = F_P * W \qquad , \qquad F'_C = F_C * W \tag{10}$$

Afterward, the cosine similarity between the F'_P and F'_C is calculated for the parent and child pair:

$$Cosine\_Similarity(F'_P, F'_C) = \frac{F'_P \cdot F'_C}{\|F'_P\| \cdot \|F'_C\|} \tag{11}$$

where $(F'_P \cdot F'_C)$ shows the inner product of the two feature vectors and $\|F\|$ indicates the Euclidean norm of F. Finally, the kinship relationship is verified by comparing the cosine similarity with a pre-defined threshold.

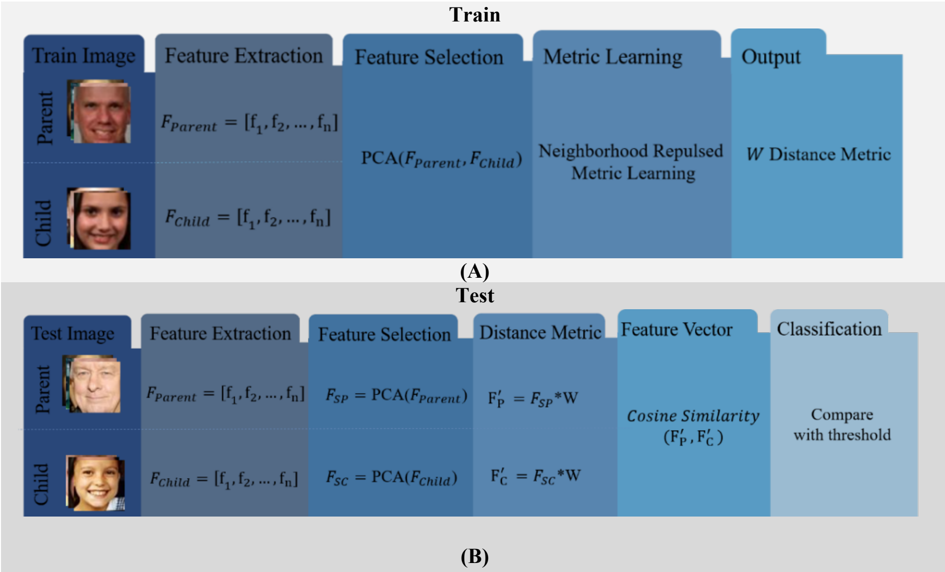


Figure 3: General trend of the first proposed method - (a): training the W distance metric and (b): evaluation of the proposed method

- Classification on metric learning similarity output

In the previous section, the final classification is performed using a thresholding strategy, as proposed by [Lu et al., 2012]. In the second proposed method, an SVM-based classifier is trained over the similarity output of the NRML method to automatically classify the kinship relations. The training stage of the SVM classifier is shown in [Fig. 4]. At first, the W distance function is learned using the training samples. To train the SVM classifier, the feature vectors (F_P *and* F_C) are mapped by W into new feature space (F'_P *and* F'_C). Then, the SVM classifier is trained over the cosine similarity of the parent/child pairs.

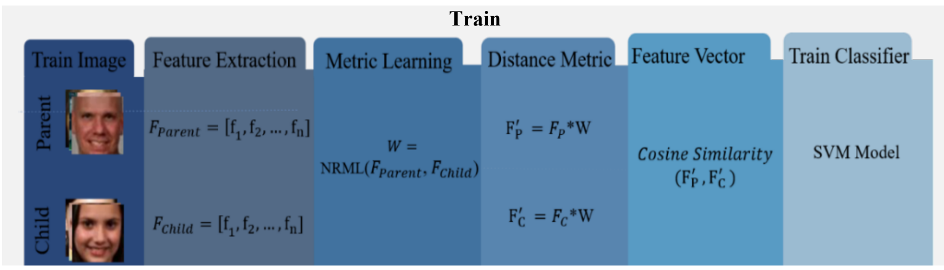
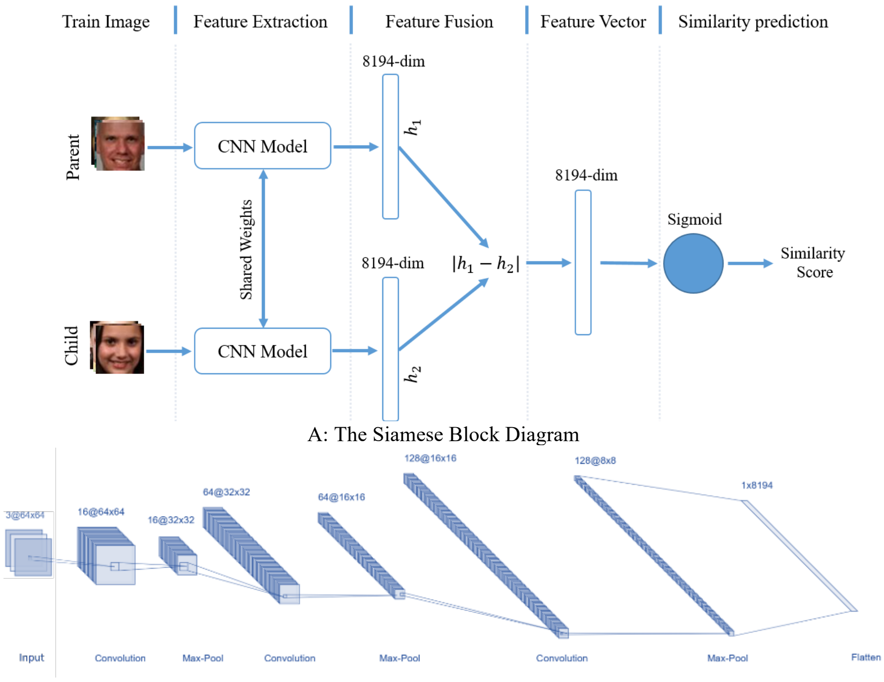


Figure 4: General trend of training the second proposed method

Finally, for the test stage, the test samples are classified similar to [Fig. 3-B] but using the SVM classifier instead of the thresholding method.

- Siamese Convolutional Neural Network

Here, the kinship verification is also performed based on deep features extracted by a Siamese convlution neural network [Koch et al., 2015]. The overall structure of the designed network is shown in [Fig. 5-A]. The network has two identical CNN models which share weights. The parent and child images are simultaneously given to the two branches. The output of each branch is a vector of size 8194. The difference between the two vectors is considered as the feature vector of the two input images. It is fed into a dense layer to classify the paired-sample and verify their relationship. The structure of the CNN models is also shown in [Fig. 5-B]. Both CNN models have the same stuructres with the same parameteres and weights.



A: The Siamese Block Diagram

B: The CNN Model Architecture

Figure 5: General structure of the proposed neural network - (A): Siamese Block Diagram (B): CNN Model Architecture

RESULTS AND EVALUATION OF THE PROPOSED APPROACH

The performance of the proposed methods is evaluated on two well-known kinship verification datasets. In this section, the datasets and evaluation criteria are explained. Then, the practical details are provided on feature extraction, feature fusion, and feature parameters. Finally, the results of the proposed method are studied in different conditions and compared to previous researches.

A. Benchmark datasets

KinFaceW-I and KinFaceW-II are two well-known kinship verification datasets [Lu et al., 2012]. These datasets include images of four parent-child relationships: father-son

(F-S), father-daughter (F-D), mother-son (M-S), and mother-daughter (M-D). For these relations, the KinFaceW-I dataset has156, 134, 116, and 127 pairs of parent-child relationships, respectively. Also, the KinFaceW- II dataset has 250 pairs of kinship images for each relationship. [Fig. 6] shows an example of positive (samples with kin relation) and negative (samples without kin relation) images for these datasets.

It should be noted that using the other relationships such as sister-brother, grandfather-grandson and etc. could be useful to improve our kinship verification system. However, the benchmark datasets used in this research only contain first-degree relationships including F-D, F-S, M-D, and M-S; they have no data for second- or higher-degree relationships. In these datasets, the pair images of different parents and children without any kinship relationships are used as the negative samples.

B. Evaluation criteria

The KinFaceW-I and KinFaceW-II datasets have been split into five-folds, each containing an equal number of positive and negative pair images. Therefore, we use 5-fold cross-validation to analyze the performance of the methods. Same as the previous works, we use accuracy metric to compare the methods with each other. The accuracy is calculated as:

$$Accuracy = \frac{P_{Correct}}{P_{Total}} \qquad (12)$$

where P_{Total} indicates the total number of paired-samples and $P_{Correct}$ shows the number of correctly classified paired samples. Suppose a paired-sample (x,y) show a (parent, children) relationship. The label of this sample is 1 when x and y really have relationship; and otherwise, the label is 0. This sample is classified as correct when we correctly predict the sample label.

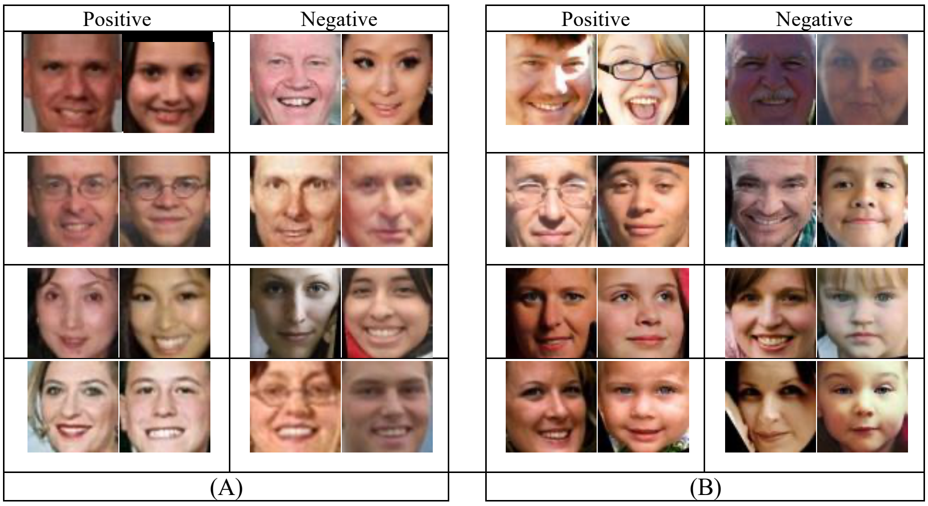


Figure 6: Example of positive and negative images of KinFaceW-I (a) and KinFaceW-II (b) databases. From top to bottom, the rows represent the father-daughter (F_D), father-son (F_S), mother-daughter (M_D), and mother-son (M_S) relationships.

C. Implementation details

This section provides additional information about feature extraction methods and their corresponding parameters. It should be noted that principal component analysis is utilized to reduce the number of features [Lu et al., 2015] in all experiments.

• For the LBP descriptor, each image is divided into 8×8 non-overlapping blocks with the size of 8×8. Finally, the LBP feature is extracted with a 59-bin histogram from each block [Lu et al., 2015].

• For the HOG descriptor, the image is divided into 16×16 (and 8x8) non-overlapping blocks with the size of 4×4 (and 8x8). The HOG is extracted with a 9-bin histogram from each block [Lu et al., 2015].

• The Binarized Statistical Image Feature (BSIF) is extracted from the image in different color spaces, including RGB, HSV, and grey images. According to [Fig. 7], this feature extraction is analyzed in two cases, (1) whole image, (2) block-wise image.

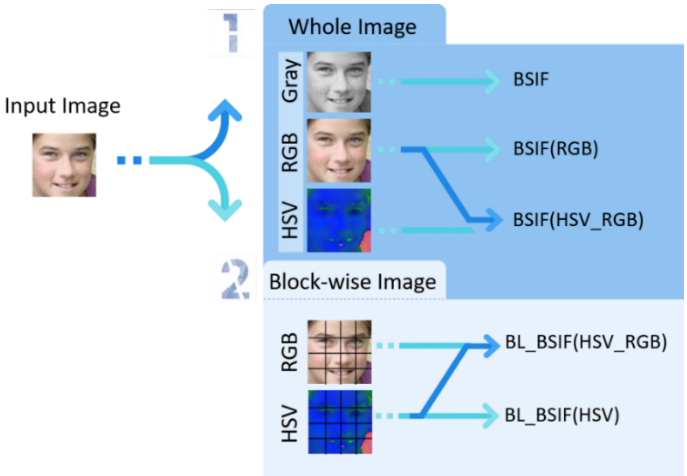


Figure 7: BSIF feature vectors in three modes

In the first case, the BSIF feature has been extracted from the grayscale, RGB, and RGB-HSV images. These feature vectors are named BSIF, BSIF (RGB), and BSIF (HSV_RGB), respectively. For the second case, the BSIF features are extracted from the same size non-overlapping blocks, labeled as BL_BSIF (HSV) and BL_BSIF (HSV_RGB), respectively.

- For PML_RCM, we have proposed to use a new mapping function instead of the mapping function proposed by Moujahid [Moujahid et al., 2019] in equation 4. The new mapping function calculated by:

$$\emptyset(\mathrm{I},\mathrm{x},\mathrm{y}) = \begin{bmatrix} x \quad y \quad R(x,y) \quad G(x,y) \quad B(x,y) \quad H(x,y) \quad S(x,y) \quad V(x,y) \\ |\mathrm{I_x}| \quad |\mathrm{I_y}| \quad |\mathrm{I_{xy}}| \quad |\mathrm{I_{yx}}| \quad |\mathrm{I_{xx}}| \quad |\mathrm{I_{yy}}| \\ LBP_{u2} \quad LBP_{ri} \quad LBP_{riu2} \quad I(x,y) \quad \theta(x,y) \end{bmatrix} \tag{13}$$

where the parameters are defined as before. The parameter $\theta(x.y)$ shows the edge orientation of pixel (x, y) calculated as:

$$\theta(x,y) = \arctan\left(\frac{|I_y|}{|I_x|}\right) \tag{14}$$

- PML_GRCM

The new PML_GRCM descriptor is defined by fusing the Multilevel Pyramid Descriptor (PML) and the GRCM feature. As such, for blocks of each pyramid, GRCM feature vector is extracted based on the mapping function 6.

The number of pyramid levels and block size for PML descriptors (ℓ, b), is sat (4,16), (7,9), and (4,16) for RCM1 ،PML_RCM2 and PML_GRCM features.

- QLRBP

Finally, RGB and HSV color spaces are used to extract RGB_QLRBP and HSV_QLRBP feature vectors. The reference quaternions are considered as (1,0,0), (0,1,0) and (0,0,1). Also, CTQ phase weights (α_1 ، α_2 and α_3) are set to 1.

D. Result analysis and evaluation

The proposed method is analyzed in three sections. In the first section, the results of the NRML metric learning method are given for texture and color features and their combination. In the second section, the results obtained from the NRML metric learning and SVM classifier method are evaluated. Also, by analyzing the various features, the best combination of features has been selected. In the third section, the best results of the proposed method are compared with the results of the previous studies.

1) Evaluate the NRML metric learning and feature fusion

In this section, first proposed method has been evaluated by various set of features and their combination. These features include the following: HOG ،BSIF(HSV) ،BSIF(HSV_RGB) ،BL_BSIF(HSV_RGB) ،REG_BSIF ، PML_GRCM ،REG_RCM ،PML_RCM1 ،PML_RCM2 ،RGB_QLRBP and HSV_QLRBP.

The results obtained for each feature and their most suitable combinations are given in [Tab. 1] and [Tab. 2] for the two KinFaceW-I and KinFaceW-II databases. In these tables, similar to the previous researches the accuracy of each relation (F-D, F-S, M-D, M-S) is calculated individually and the mean accuracy is also reported.

Name	Method	Classifier	Feature	F-D	F-S	M-D	M-S	Mean
M1	NRML	-	BSIF(HSV)	66	67.9	70.9	67.8	68.1
M2	NRML	-	BSIF(HSV_RGB)	66	68.9	70.1	68	68.3
M3	NRML	-	PML_GRCM	64.5	70.4	69.6	63.7	67.12
M4	NRML	-	BL_BSIF(HSV_RGB)	67.1	72.1	70.8	65	68.8
M5	NRML	-	PML_RCM1	69.4	72	75.3	71.5	72.09
M6	NRML	-	PML_RCM1+PML_GRCM	69.4	73.3	76.1	70.6	72.38
M7	NRML	-	BL_BSIF(HSV)	69.4	77.8	73.1	70.6	72.7
M8	NRML	-	RGB_QLRBP+ HSV_QLRBP	70.8	75.9	75.4	69.4	72.9
M9	NRML	-	RGB_QLRBP	70.8	75.6	76.6	70.2	73.3
M10	NRML	-	HOG+ RGB_QLRBP+ HSV_QLRBP	71.6	76.9	75.9	71.1	73.9
M11	NRML	-	HSV_QLRBP	71.6	77.9	75.1	71.6	74
M12	NRML		PML_RCM1+PML_RCM2	70.9	73.6	76.9	74.5	74
M13	NRML	-	PML_RCM1+ PML_RCM2+PML_GRCM	71.6	74.3	78.4	71.5	74

M14	NRML	-	RGB_QLRBP+PML_RCM1+ PML_RCM2+PML_GRCM	71.6	74.3	76.5	74.1	74.1
M15	NRML	-	PML_RCM2+PML_GRCM	71.6	74.3	78.4	73.6	74.54
M16	NRML	-	PML_RCM2	70.9	74.3	77.7	75.4	74.59
M17	NRML	-	HOG+BL_BSIF(HSV_RGB)	73.9	82.3	79.1	71.9	76.8
M18	NRML	-	HOG+BSIF(HSV)	74.2	83.6	79.5	71.1	77.1
M19	NRML	-	HOG+BL_BSIF(HSV)	75	83	80	70	77.2
M20	**NRML**	**-**	**HOG+BSIF(HSV_RGB)**	**75**	**83.3**	**80.7**	**71.5**	**77.6**

Table 1: Results of the first proposed method for the KinFaceW-I *database*

Name	Method	Classifier	Feature	F-D	F-S	M-D	M-S	Mean
M3	NRML	-	PML_GRCM	67	68	66	67.2	67
M4	NRML	-	BL_BSIF(HSV_RGB)	67.2	72.2	70.8	73.8	71
M2	NRML	-	BSIF(HSV_RGB)	67.8	75	73.6	74.6	72.7
M1	NRML	-	BSIF(HSV)	67.6	74.8	74	75.6	73
M7	NRML	-	BL_BSIF(HSV)	68.8	77.4	73	74.6	73.4
M11	NRML	-	HSV_QLRBP	70.4	78.8	71.8	79	75
M15	NRML	-	PML_RCM2+PML_GRCM	73	72.4	80.6	77.4	75
M9	NRML	-	RGB_QLRBP	70.8	81.8	73	77.2	75.7
M16	NRML	-	PML_RCM2	72.8	72.8	80	77.4	75.75
M8	NRML	-	RGB_QLRBP+ HSV_QLRBP	71.4	81.6	73.4	78.2	76.1
M10	NRML	-	HOG+ RGB_QLRBP+ HSV_QLRBP	71.4	82.4	72.8	79	76.4
M18	NRML	-	HOG+BSIF(HSV)	73.4	82.6	73.8	76.2	76.5
M19	NRML	-	HOG+BL_BSIF(HSV)	73	83	73.4	77.8	76.8
M20	NRML	-	HOG+BSIF(HSV_RGB)	74	83.6	75.6	77.2	77.6
M17	NRML	-	HOG+BL_BSIF(HSV_RGB)	73.2	83.4	74.6	79.4	77.65
M5	NRML	-	PML_RCM1	70.6	80.4	79.2	81	77.8
M6	NRML		PML_RCM1+PML_GRCM	72.6	81.6	79.6	81.6	78.8
M12	NRML	-	PML_RCM1+PML_RCM2	72.4	80	85.8	85.2	80.8
M13	NRML	-	PML_RCM1+ PML_RCM2+PML_GRCM	73.4	80.4	86	85.4	81.3
M14	NRML	-	**RGB_QLRBP+PML_RCM1+ PML_RCM2+PML_GRCM**	**72.6**	**82**	**86.2**	**85.4**	**81.5**

Table 2: Results of the first proposed method for the KinFaceW-II *database*

The results of [Tab. 1] and [Tab. 2] show that the combination of features extracted from RGB, HSV, and grayscale color spaces, generally outperforms the features extracted from a color space. The results also show that by fusing the features extracted block-Wise from the image with each other or with the features extracted from the whole image, the accuracy of the constructed feature vector is generally better than the accuracy of each the features. According to the obtained results, the combination of HOG and BSIF(HSV_RGB) features for the KinFaceW-I database has better average accuracy. In this feature vector, the HOG feature is extracted from the image in a block-wise manner and BSIF(HSV_RGB) is extracted from the whole image. Also, the HOG feature is extracted in grayscale, and BSIF(HSV_RGB) is extracted in RGB and HSV color spaces.

For the KinFaceW-II database, the combination of RGB_QLRBP, PML_RCM1, PML_RCM2, and PML_GRCM features had the best accuracy on average. The

RGB_QLRBP feature is extracted from the whole image in RGB color space. features PML_RCM1 PML_RCM2 and PML_GRCM are extracted from the image in a block-wise manner, in three grayscale, RGB, and HSV color spaces.

2) Evaluate the NRML metric learning and SVM classifier method

Earlier in Section 3, the feature fusing, using the NRML metric learning method and the SVM classifier was discussed. To evaluate this proposed method, a different set of features are considered and the results obtained from each feature and the most appropriate feature fusion vector are shown in [Tab. 3] and [Tab. 4] for KinFaceW-I and KinFaceW-II databases.

Name	Method	Classifier	Feature	F-D	F-S	M-D	M-S	Mean
M21	NRML	SVM	BL_BSIF(HSV)	51.8	59.6	55.9	53.4	55.2
M22	NRML	SVM	BL_BSIF(HSV_RGB)	61.1	60.8	61.4	55.5	59.7
M23	NRML	SVM	PML_GRCM	60.4	62.4	65.4	58.1	61.63
M24	NRML	SVM	BSIF(HSV)	58.2	64.1	65.7	59	61.8
M25	NRML	SVM	BSIF(HSV_RGB)	61.2	64.1	63.7	60.7	62.4
M26	NRML	SVM	PML_RCM1	66	68.8	68.7	63.7	66.86
M27	NRML	SVM	RGB_QLRBP	64.2	72.1	69.2	64.6	67.5
M28	NRML	SVM	PML_RCM2	6.4	67.8	72.1	68	67.89
M29	NRML	SVM	RGB_QLRBP+PML_RCM1+ PML_RCM2+PML_GRCM	63.8	68.2	71.8	68	67.9
M30	NRML	SVM	PML_RCM2+PML_GRCM	64.2	68.2	72.2	68	68.18
M31	NRML	SVM	RGB_QLRBP+ HSV_QLRBP	66	72.7	69.2	64.6	68.19
M32	NRML	SVM	PML_RCM1+PML_RCM2	65.6	68.2	71.8	67.2	68.2
M33	NRML	SVM	PML_RCM1+PML_GRCM	64.5	69.8	72.2	66.7	68.35
M34	NRML	SVM	HSV_QLRBP	67.9	71.5	67.2	66.8	68.39
M35	NRML	SVM	HOG+ RGB_QLRBP+ HSV_QLRBP	66.4	74	69.2	63.8	68.4
M36	NRML	SVM	PML_RCM1+PML_RCM2+ PML_GRCM	66.7	71.1	74.5	63.3	68.96
M37	NRML	SVM	HOG+BSIF(HSV_RGB)	69	80.1	73.6	65.9	72.2
M38	NRML	SVM	HOG+BL_BSIF(HSV)	70.5	79.8	75.2	65	72.6
M39	NRML	SVM	HOG+BSIF(HSV)	70.9	79.2	74.4	66.3	72.7
M40	**NRML**	SVM	**HOG+BL_BSIF(HSV_RGB)**	**69.8**	**79.4**	**75.5**	**66.3**	**72.8**

Table 3: Results of the second proposed method for the KinFaceW-I *database*

Name	Method	Classifier	Feature	F-D	F-S	M-D	M-S	Mean
M21	NRML	SVM	BL_BSIF(HSV)	53	60.4	63	65	60.3
M23	NRML	SVM	PML_GRCM	63.4	63.4	61.8	62.8	62.8
M22	NRML	SVM	BL_BSIF(HSV_RGB)	64.4	64.6	62.8	68.8	65.1
M34	NRML	SVM	HSV_QLRBP	67.4	72.4	68.6	54.6	65.7
M24	NRML	SVM	BSIF(HSV)	64.6	70.8	71.6	73.2	70
M25	NRML	SVM	BSIF(HSV_RGB)	64.4	72.4	71.2	72	70
M31	NRML	SVM	RGB_QLRBP+ HSV_QLRBP	68.2	78.2	68.8	68.6	70.9
M30	NRML	SVM	PML_RCM2+PML_GRCM	68.6	67.8	75.6	73.6	71.4
M35	NRML	SVM	HOG+ RGB_QLRBP+ HSV_QLRBP	69	77.8	70	69	71.45
M28	NRML	SVM	PML_RCM2	68.4	68.4	76	73.4	71.5
M27	NRML	SVM	RGB_QLRBP	67.4	77.8	69	73.4	71.9
M39	NRML	SVM	HOG+BSIF(HSV)	69.2	81.2	68.4	73.2	73
M38	NRML	SVM	HOG+BL_BSIF(HSV)	67.8	81.4	69.6	74.2	73.2
M37	NRML	SVM	HOG+BSIF(HSV_RGB)	69.2	81.4	70.8	73.6	73.7

M26	NRML	SVM	PML_RCM1	68.4	77	74.4	77	**74.2**
M40	NRML	SVM	HOG+BL_BSIF(HSV_RGB)	69.6	80.6	71.2	78	74.8
M33	NRML	SVM	PML_RCM1+PML_GRCM	69.2	79	75.2	79.2	75.6
M32	NRML	SVM	PML_RCM1+PML_RCM2	68.2	77.6	82.6	82.2	77.6
M36	NRML	SVM	PML_RCM1+PML_RCM2+ PML_GRCM	69	77.4	83	82.6	78
M29	NRML	SVM	**RGB_QLRBP+PML_RCM1+ PML_RCM2+PML_GRCM**	**69.2**	**79.4**	**81.6**	**82.4**	**78.1**

Table 4: Results of the second proposed method for the KinFaceW-II *database*

As the evaluation results showed in the previous section, the combination of features in RGB, HSV, and grayscale color spaces and fusing the features extracted from the whole image or extracted block-Wise from the image outperforms the features extracted from a color space. Results show that, the combination of HOG and BL_BSIF(HSV_RGB) features for the KinFaceW-I database has better average accuracy. the HOG and BL_BSIF(HSV_RGB) features are extracted from the image in a block-wise manner. Also, the HOG feature is extracted in grayscale, and BL_BSIF(HSV_RGB) is extracted in RGB and HSV color spaces.

Like pervious proposed method, the combination of RGB_QLRBP, PML_RCM1, PML_RCM2, and PML_GRCM features had the best average accuracy for the KinFaceW-II database.

3) Evaluate the Siamese Convolutional Neural Network

The Siamese network model is trained using the structure described earlier in the previous section. The method is separately evaluated on each kinship relationship. [Tab. 5] shows the results for the two datasets KinFaceW-I and KinFaceW-II. The results show that the deep features obtained by Siamese neural network have accuracies of 73 and 82 respectively for the two KinFaceW-I and KinFaceW- II datasets. In next section, the results are compared with the other proposed methods and other researches.

Dataset	Name	Method	Classifier	Feature	F-D	F-S	M-D	M-S	Mean
KinFaceW-I	**M41**	Siamese CNN	-	-	71	69	80	71	73
KinFaceW-II	**M41**	Siamese CNN	-	-	80	79	86	82	82

Table 5: Results of the third proposed method for the KinFaceW-I and KinFaceW- II database

4) Comparison with previous methods

As the results of the proposed method in the previous sections showed, the M40, M20, M29, M14, and M41 feature vectors performed better for the KinFaceW-I and KinFaceW-II databases.

In this section, to evaluate the proposed methods, the feature vectors mentioned are compared with other previous methods. This comparison is given in [Tab. 6] and [Tab. 7] for the KinFaceW-I and KinFaceW- II databases.

Name	Method	Classifier	Feature	F-D	F-S	M-D	M-S	Mean
Moujahid et] [al., 2019	PML	SVM	COV	50.7	50	50.3	48.2	49.8
Goyal et al.,] [2020	NCC	-	-	54.2	56.1	56.2	54.4	55.23
[Yan, 2017]	NRCML	-	LE	61.1	**66.1**	**73**	66.9	66.3
[Lu et al., 2012]	MNRML	SVM	LBP + TPLBP + SIFT + LE	66.5	72.5	72.0	66.2	69.9
[Lan et al., 2016]	QMD	-	QMCBP	68.1	**72.2**	**76**	67.8	71
[López et al., 2016]	Simple scoring	-	-	65.7	**69.9**	**79.6**	70.7	71.4
[Wu et al., 2018]	Shallow	ELM	BSIF	64.2	70.0	77.2	73.0	71.7
[Li et al., 2016]	SMCNN	-	-	75	**75**	**72.2**	68.7	72.2
[Van et al., 2019]	-	SVM	LBP	69.2	78.1	72.2	70.8	72.6
M40	NRML	SVM	HOG+BL_BSIF(HSV_RGB)	69.8	79.4	75.5	66.3	72.8
M41	Siamese CNN	-	-	71	69	80	71	73
Wu et al.,] [2021	CML	-	SIFT	71.15	71.33	76.8	75.9	73.8
Zhang et al.,] [2015	CNN-Points	-	-	71.8	76.1	84.1	78	77.5
Ravi Kumar] [et al., 2020	NRML	-	WFBT-SBP	77.33	77.4	77.94	77.98	77.6
M20	NRML	-	HOG+BSIF(HSV _RGB)	75	83.3	80.7	71.5	77.6
[Wang et al., 2020]	NESN-KVN	-	-	76.8	77	85.2	75.8	78.6
[Li et al., 2020]	GKR	-	-	79.5	73.2	78	86.2	79.2
Dornaika et] [al., 2020	VGG-Face - VGG-F	-	-	79.85	85.9	86.62	86.2	84.55
Yan et al.,] [2021	MSDR	-	-	85.8	87.5	88.1	80.9	85.6
Laiadi et al.,] [2020	MSIDA+W CCN	-	-	85.98	85.93	90.05	88.62	87.65

Table 6: Comparison of proposed methods with previous methods for the KinFaceW-I database

Name	Method	Classifier	Feature	F-D	F-S	M-D	M-S	Mean
Moujahid et al.,] [2019	PML	SVM	COV	51.4	51.4	50.6	50.8	51
[Goyal et al., 2020]	NCC	-	-	62.49	47.33	56.8	46.8	53.38
[Lu et al., 2015]	NRML	-	HOG	72	80	70	74	74
[Bisogni et al., 2022]	SNN	-	-	-	-	-	-	75
[Lan et al., 2016]	QMD	-	QMCBP	71.6	**77.2**	**73.4**	79	75.3
[Zhou et al., 2016]	ESL	-	HOG	73	**81.2**	**73**	75.6	75.7
[Zhou, Yan and Shang, 2016]	Multivi ew SSL	-	HOG, LBP	74	**81.8**	**72.5**	75.3	75.9
[Lu et al., 2012]	MNRM L	SVM	LBP + TPLBP + SIFT + LE	74.3	76.9	77.6	77.4	76.5
[Lan et al., 2017]	QIWLD	-	QWLD	73.6	**77.4**	**76.8**	78.4	76.6

[Ravi Kumar et al., 2020]	NRML	-	WFBT-SBP	76.72	**76.8**	**76.9**	77	76.85
M29	NRML	SVM	RGB_QLRBP+PML_RCM1+ PML_RCM2+PML GRCM	69.2	79.4	81.6	82.4	78.1
[Wu et al., 2021]	CML	-	SIFT	78.3	80	78.3	76.2	78.2
M14	NRML	-	RGB_QLRBP+PML_RCM1+ PML_RCM2+PML GRCM	72.6	82	86.2	85.4	81.5
M41	Siamese CNN	-	-	80	79	86	82	82
[Dornaika et al., 2020]	VGG-Face _ VGG-F	-	-	82.60	87.2	89.40	88.40	86.90
[Laiadi et al., 2020]	MSIDA +WCCN	-	-	89.40	**82.80**	**87.80**	**88.00**	87.00
[Zhang et al., 2015]	CNN-Points	-	-	81.9	89.4	92.4	89.9	88.4
[Yan et al., 2021]	MSDR	-	-	90.6	86.6	91	87.2	88.8
[Wang et al., 2020]	NESN-KVN	-	-	86.7	88.7	91.6	89.1	89.0
[Li et al., 2020]	GKR	-	-	90.8	86.0	91.2	94.4	90.6

Table 7: Comparison of proposed methods with previous methods for the KinFaceW-II database

The results of the KinFaceW-I database in [Tab. 6] show that the proposed M40 and M20 methods can improve the kinship verification system performance by the feature fusion method, rather than the previous methods [Wu et al., 2018], [Van et al., 2019], [Yan, 2017], and [Lan et al., 2016] that use a single feature. By comparing the methods [Wu et al., 2018], [Van et al., 2019], [Moujahid et al., 2019] and the proposed M40 method, it is clear that the use of a classifier for metric learning similarity can improve the performance of classifier models. In addition, method M20 shows that the feature fusion along with metric learning has performed better than both classifier methods and other metric methods, like [Lu et al., 2012], [Yan, 2017], and [Wu et al., 2021]. Also, according to the results, NRML metric learning has the same average accuracy for both M20, [Zhang et al., 2015] and [Ravi Kumar et al., 2020] methods. Although, the proposed method has performed relatively better in F-S and M-D verification. the proposed M40 and M20 methods have better results than the method [López et al., 2016], that used the neural network.

According to [Tab. 7], the proposed M14 and M29 methods for the KinFaceW-II database have better results than the other methods. As mentioned before, in the proposed methods, with the help of the feature fusion, the average accuracy is better than the previous methods [Lan et al., 2016], [Zhou et al., 2016], [Lan et al., 2017], and [Lu et al., 2015] that have used one feature. also, the M29 method outperforms the two NRML based methods [Lu et al., 2015] and [Ravi Kumar et al., 2020].

The performance of the Siamese network with 82% accuracy on KinFaceW-II is comparable to our best metric learning with 81.5% accuracy. It is while, the metric learning method with 77.6% accuracy on KinFaceW-I, achieves higher performance than the Siamese network with 73% accuracy. It might be due to lack of enough data for training the network on KinFaceW-I dataset which has less number of samples respect to KinFaceW-II. The results [Wang et al., 2020, Zhang et al., 2015, Laiadi et al., 2020, Li et al., 2020, Dornaika et al., 2020] show that deeper neural network, which benefit from transfer learning or data augmentation, led to better results, but at some expense. They have high number of hyper-parameters should be tuned via trial- and-error experiments [Yan et al., 2021, Wang et al., 2020]. Besides, they have high computational complexity and they are usually implemented on GPU platforms at an expensive cost [Zhou et al., 2021, Armeniakos et al., 2022, Yu et al., 2021]. In the conditions that these operational costs can be afford, deep learning solution is a good alternative; Otherwise, the classical methods can be helpful.

The results of the F_D, F_S, M_D, and M_S relationships in both databases show the effect of gender on the kinship verification system. Thus, the results of same-gender F_S and M_D kinship relationships are better than F_D and M_S relationships in which gender is different. Therefore, gender in the facial image pair can affect the final result. Finally, it can be concluded that the use of metric learning methods to find appropriate metrics from training data, and classification models for automatic kinship verification, was associated with good results. The results of the proposed method also showed that the fusion of features extracted from RGB, HSV color spaces, and grayscale level, as well as features extracted from blocks and the whole image, can increase the average accuracy.

4 CONCLUSION

This paper presents an efficient kinship verification system. In this approach, color and texture features were first merged. Then, NRML metric learning method and kinship classification using a SVM classifier is proposed. Also, a Siamese convolutional neural network was presented for kinship detection. The approaches are evaluated by two databases, KinFaceW-I and KinFaceW-II. The results showed that the features fusion can improve the final results of kinship verification based on metric learning. It was also found that by fusing features in different color spaces such as RGB, HSV, and grayscale, as well as feature extraction at different block levels and the whole image, a more discriminative feature vector could be created. In addition, the results of the Siamese network showed that its performance can be comparable with the methods based on metric learning when enough data is avilable. The racial difference between parents and children such as skin color variations can affect the performance of a kinship verification system. It would be beneficial to study this challenge as the future work. Besides, in the continuation of this research, we plan to focus more on deep learning methods in conjunction with the results obtained from the current research.

References

[Agbo-Ajala et al., 2021] Agbo-Ajala, O., Viriri, S.: "Deep learning approach for facial age classification: a survey of the state-of-the-art"; *Artificial Intelligence Review volume,* 54, (2021), 179–213.

[Alghaili et al., 2020] Alghaili, M., Li, Z., Ali, A.: "Deep feature learning for gender classification with covered/camouflaged faces"; *IET Image Processing,* 14, 15 (2020), 3957 – 3964.

[Armeniakos et al., 2022] Armeniakos, G., Zervakis, G., Soudris, D., Henkelel, J. : *"Hardware Approximate Techniques for Deep Neural Network Accelerators: A Survey"*; ACM Computing Surveys, (2022).

[Bellet et al., 2013] Bellet, A., Habrard, A., Sebban, M.: *"A Survey on Metric Learning for Feature Vectors and Structured Data"*; in ArXiv, (2013).

[Bisogni et al., 2022] Bisogni, C., Narducci, F.: "Kinship recognition: how far are we from viable solutions in smart environments?"; *Procedia Computer Science,* 198, (2022), 225-230.

[Dagher et al., 2021] Dagher, I., Barbara, D.: "Facial age estimation using pre-trained CNN and transfer learning"; *Multimedia Tools and Applications,* 80, (2021), 20369–20380.

[Dornaika et al., 2019] Dornaika, F., Arganda-Carreras, I., Serradilla, O.: *"Transfer learning and feature fusion for kinship verification"*; Neural Computing and Applications, (2019), 1–13.

[Dornaika et al., 2020] Dornaika, F., Arganda-Carreras, I., Serradilla, O.: *"Transfer learning and feature fusion for kinship verification"*; Neural Computing and Applications, 32, (2020), 7139–7151.

[Goyal et al., 2020] Goyal, A., Meenpal, T.: "Template Matching for Kinship Verification in the Wild"; in Computational Intelligence in Pattern Recognition. Advances in Intelligent Systems and Computing, Singapore, (2020).

[Guo et al., 2022] Guo, X., Wei, W., Liang, J., Dang, C., Liang, J.: *"Metric learning via perturbing hard-to-classify instances"*; Pattern Recognition, 132, (2022), 108928.

[Kale et al., 2021] Kale, A., Altun, O.: "Age, Gender and Ethnicity Classification from Face Images with CNN-Based Features"; in *Innovations in Intelligent Systems and Applications Conference (ASYU)*, Elazig, (2021).

[Kannala et al., 2012] Kannala, J., Rahtu, E.: *"BSIF: Binarized Statistical Image Features"*; in Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, (2012).

[Kim, 2021] Kim, P. W.: "Image super-resolution model using an improved deep learning-based facial expression analysis"; *Multimedia Systems,* 27, 4 (2021), 615–625.

[Koch et al., 2015] Koch, G., Zemel, R., Salakhutdinov, R.: *"Siamese Neural Networks for One-shot Image Recognition"*; International Conference on Machine Learning, Lille, France, (2015), 1–8.

[Laiadi et al., 2019] Laiadi, O., Ouamane, A., Benakcha, Taleb-Ahmed, Hadid, A.: "Kinship Verification based Deep and Tensor Features through Extreme Learning Machine"; 14th IEEE International Conference on Automatic Face & Gesture Recognition , Lille, France, (2019).

[Laiadi et al., 2019] Laiadi, O., Ouamane, A., Boutellaa, E., Benakcha, A., Taleb-Ahmed, A., Hadid, A.: *"Kinship verification from face images in discriminative subspaces of color components"*; Multimedia Tools and Applications, 78, 12 (2019), 16465–16487.

[Laiadi et al., 2020] Laiadi, O., Ouamane, A., Benakcha, A., Taleb-Ahmed, A., Hadid, A.: *"Multi-view Deep Features for Robust Facial Kinship Verification"*; IEEE International Conference on Automatic Face and Gesture Recognition(IEEE AMFG), (2020), 877-881.

[Lan et al., 2016] Lan, R., Zhou, Y., Tang, Y. Y.: *"Quaternionic Local Ranking Binary Pattern: A Local Descriptor of Color Images"*; IEEE Transactions on Image Processing, 25, 2 (2016), 566-579.

[Lan et al., 2016] Lan, R., Zhou, Y.: *"Quaternion-Michelson Descriptor for Color Image Classification"*; IEEE Transactions on Image Processing, 25, 11 (2016), 5281-5292.

[Lan et al., 2017] Lan, R., Zhou, Y., Tang, Y. Y.: *"Quaternionic Weber Local Descriptor of Color Images"*; IEEE Transactions on Circuits and Systems for Video Technology, 27, 2 (2017), 261-274.

[Li et al., 2016] Li, L., Feng, X., Wu, X., Xia, Z., Hadid, A.: *"Kinship Verification from Faces via Similarity Metric Based Convolutional Neural Network"*; International conference image analysis and recognition. Springer, 9730, (2016), 539–548.

[Li et al., 2020] Li, W., Zhang, Y., Lv, K., LU, J., Feng, J., Zhou, J.: *"Graph-Based Kinship Reasoning Network"*; *IEEE International Conference on Multimedia and Expo (ICME)*, (2020), 1-6.

[Liu et al., 2022] Liu, D., Wu, L., Hong, R., Ge, Z., Shen, J., Boussaid, F., Bennamoun M. : *"Generative Metric Learning for Adversarially Robust Open-World Person Re-Identification"*; *ACM Transactions on Multimedia Computing, Communications, and Applications*, (2022).

[López et al., 2016] López, M. B., Boutellaa, E., Hadid, A.: *"Comments on the "Kinship Face in the Wild" Data Sets"*; IEEE Transactions on Pattern Analysis and Machine Intelligence, 38, 11 (2016), 2342-2344.

[Lopez et al., 2018] Lopez, M. B., Hadid, A., Boutellaa, E., Goncalves, J., Kostakos, V., S, Hosio.: *"Kinship verification from facial images and videos: human versus machine"*; Machine Vision and Applications, 29, 5 (2018), 873–890.

[Lu et al., 2012] Lu, J., Hu, J., Zhou, X., Shang, Y., Tan, Y. P., Wang, G.: *"Neighborhood repulsed metric learning for kinship verification"*; in IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, (2012).

[Lu et al., 2015] Lu, J., Hu, J., Liong, V. E., Zhou, X., Bottino, A., Islam, I.U., Vieira, I. F., Qin, X., Tan, X., Chen, S., Mahpod, S., Keller, Y., Zheng, L., Idrissi, K., Garcia, C., Duffner, S., Baskurt, A., Castrillon-Santana, M., Lorenzo-Navarro, J. : *"The FG 2015 Kinship Verification in the Wild Evaluation"*; in 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), Ljubljana, (2015).

[Moujahid et al., 2019] Moujahid, A., Dornaika, F.: *"A pyramid multi-level face descriptor: application to kinship verification"*; Multimedia Tools and Applications volume, 78, 7 (2019), 9335-9354.

[Nandy et al., 2019] Nandy, A., Mondal, S. S.: *"Kinship Verification using Deep Siamese Convolutional Neural Network"*; in 14th IEEE International Conference on Automatic Face & Gesture Recognition, Lille, France, (2019).

[Pang et al., 2008] Pang, Y., Yuan, Y., Li, X.: *"Gabor-Based Region Covariance Matrices for Face Recognition"*; IEEE Transactions on Circuits and Systems for Video Technology, 18, 7 (2008), 989-993.

[Ramazankhani et al., 2021] Ramazankhani, F., Yazdian-Dehkordi, M.: "Color and Texture Feature Fusion for Facial Kinship Verification"; Journal of Machine Vision and Image Processing (JMVIP), 9, 2 (2021), 1-17.

[Ratnaparkhi et al., 2021] Ratnaparkhi, S.T., Tandasi, A., Saraswat, S.: "Face Detection and Recognition for Criminal Identification System"; in *11th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, Noida, (2021).

[Ravi Kumar et al., 2020] Ravi Kumar, Y. B., Narayanappa, C. k., Dayananda, P.: "Weighted full binary tree-sliced binary pattern: An RGB-D image descriptor"; *Heliyon,* 6, 5 (2020).

[Robinson et al., 2018] Robinson, J. p., Shao, M., Wu, Y., Liu, H., Gillis, T., Fu, Y.: *"Visual Kinship Recognition of Families in the Wild"*; IEEE Transactions on Pattern Analysis and Machine Intelligence, 40, 11 (2018), 2624-2637.

[Royakkers et al., 2018] Royakkers, L., Timmer, J., Kool, L., Est, R.v. : ″Societal and ethical issues of digitization″; Ethics and Information Technology, 20, (2018), 127-142.

[Smith et al., 2022] Smith, M., Miller, S.: ″The ethical application of biometric facial recognition technology″; AI & SOCIETY, 37, (2022), 167-175.

[Sudhakar et al., 2021] Sudhakar, K., Nithyanandam, P.: "Facial identification of twins based on fusion score method"; *Journal of Ambient Intelligence and Humanized Computing, (2021),* 1–12.

[Swaminathan et al., 2020] Swaminathan, A., Chaba, M., Sharma, D.K., Chaba, Y.: "Gender Classification using Facial Embeddings: A Novel Approach"; Procedia Computer Science, 167 (2020), 2634-2642.

[Tang et al., 2022] Tang, J., Li, D., Tian, Y.: *"Image classification with multi-view multi-instance metric learning"*; Expert Systems with Applications, 189, (2022), 116117.

[Tuzel et al., 2006] Tuzel, O., Porikli, F., Meer, P.: *"Region Covariance: A Fast Descriptor for Detection and Classification"*; in 9th European conference on Computer Vision, Berlin, Heidelberg, (2006).

[Van et al., 2019] Van, T. V., Hoang, V. T.: "Kinship Verification based on Local Binary Pattern features coding in different color space"; in 26th International Conference on Telecommunications (ICT), Hanoi, Vietnam, (2019).

[Wang et al., 2019] Wang, S., Ding, Z., Fu, Y.: *"Cross-Generation Kinship Verification with Sparse Discriminative Metric"*; IEEE Transactions on Pattern Analysis and Machine Intelligence, 41, 11 (2019), 2783-2790.

[Wang et al., 2020] Wang, S., Yan, H.: *"Discriminative sampling via deep reinforcement learning for kinship verification"*; Pattern Recognition Letters, 138, (2020), 38-43.

[Wu et al., 2018] Wu, X., Feng, X., Boutellaa, E., Hadid, A.: *"Kinship Verification using Color Features and Extreme Learning Machine"*; in IEEE 3rd International Conference on Signal and Image Processing (ICSIP), (2018).

[Wu et al., 2018] Wu, Y., Liu, H., Robinson, J., Fu, Y.: *"Kinship Classification through Latent Adaptive Subspace"*; in 13th IEEE International Conference on Automatic Face & Gesture Recognition, Xi'an, (2018).

[Wu et al., 2019] Wu, X., Granger, E., Feng, X., Kinnunen, T. H., Hadid, A.: *"Audio-Visual Kinship Verification in the Wild"*; Computing Research Repository (CoRR), abs/1906.10096, (2019).

[Wu et al., 2021] Wu, H., Chen, J., Liu, X., Hu, J.: "Component-based metric learning for fully automatic kinship verification," *Visual Communication and Image Representatio,* 79, 1, (2021), 1-6.

[Yan et al., 2021] Yan, H., Song, C.: "Multi-scale deep relational reasoning for facial kinship verification"; *Pattern Recognition,* 110, (2021), 107541.

[Yan, 2017] Yan, H.: *"Kinship verification using neighborhood repulsed correlation metric learning"*; Image and Vision Computing, 60, (2017), 91-97.

[Yu et al., 2021] Yu, G. X., Gao, Y., Golikov, P., Pekhimenko, G. : *"Habitat: A Runtime-Based Computational Performance Predictor for Deep Neural Network Training"*; USENIX Annual Technical Conference (USENIX ATC 21), (2021), 503-521.

[Zhang et al., 2015] Zhang, K., Huang, Y., Song, C., Wu, H., Wang, L.: *"Kinship Verification with Deep Convolutional Neural Networks"*; Proceedings of the British Machine Vision Conference (BMVC), (2015), 1-12.

[Zhou et al., 2016] Zhou, X., Shang, Y., Yan, H., Guo, G.: *"Ensemble similarity learning for kinship verification from facial images in the wild"*; Information Fusion, 32, PB (2016), 40-48.

[Zhou et al., 2016] Zhou, X., Yan, H., Shang, Y.: *"Kinship verification from facial images by scalable similarity fusion"*; similarity fusion, 197, (2016), 136-142.

[Zhou et al., 2021] Zhou, F., Sun, T., Hu, X., Wang, T., Wen, C. : *"A sparse denoising deep neural network for improving fault diagnosis performance"*; Signal, Image and Video Processing volume, 15, (2021), 1889–1898.

[Zhu et al., 2021] Zhu, J., Shao, M., Xia, C., Pan, H., Xia,S.: *"Adversarial Attacks on Kinship Verification using Transformer"*; IEEE International Conference on Automatic Face and Gesture Recognition, Jodhpur, India, (2021), 1–8.