Computer Vision and Image Understanding 184 (2019) 9–21

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Discriminant Deep Feature Learning based on joint supervision Loss and Multi-layer Feature Fusion for heterogeneous face recognition ☆

Weipeng Hu, Haifeng Hu *

School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China

ARTICLE INFO

Communicated by N. Paragios

Keywords:
Heterogeneous face recognition
Deep learning
Joint supervision loss
Feature fusion

ABSTRACT

Heterogeneous face recognition (HFR) is still a challenging problem in computer vision community due to large appearance difference between near infrared (NIR) and visible light (VIS) modalities. Recently, breakthroughs have been made for traditional face recognition by applying deep learning on a huge amount of labeled VIS face samples. However, the same deep learning approach cannot be simply applied to HFR task due to large domain difference as well as insufficient pairwise images in different modalities during training. In general, the pooling layer of deep network can play the role of feature reduction, but also lead to the loss of useful face information, resulting in a decrease in the performance of HFR problem. It is important to eliminate modal-related information and retain more facial identity information. In this paper, we propose a novel method called Discriminant Deep Feature Learning Based on Joint Supervision Loss and Multi-layer Feature Fusion (DDFLJM) for HFR task. In most of the available CNNs, the softmax loss function is used as the supervision signal to train the deep model. In order to enhance the discriminative power of the deeply learned features, this paper proposes a new loss function called Scatter Loss (SL), which embeds both inter- and intra-class information for effectively training the deep model. To make full use of the various layers of the deep network, a Dimension Reduction Block (DRB) is designed to effectively extract the auxiliary features on multiple mid-level layers. An orthogonality constraint is introduced to the DRB block to reduce spectrum variations of two different modalities. The proposed SL is applied to multiple layers of network for joint supervision training, which enables multiple layers of the network to obtain discriminative identity features. Moreover, a Modified Gate Two-stream Neural Network (MGTNN) is adopted to fuse multiple-layer features. Extensive experiments are carried out on two challenging NIR-VIS HFR datasets CASIA NIR-VIS 2.0 and Oulu-CASIA NIR-VIS, demonstrating the superiority of the proposed method.

1. Introduction

This paper focuses on the heterogeneous face recognition (HFR) problem, which has been widely studied in the field of computer vision. HFR refers to the task which matches a probe with the gallery taken from alternate imaging modality. The major difficulties of HFR lie in the great discrepancies between different image modalities. During the last decade, many methods have been proposed to alleviate the appearance difference from heterogeneous data. Most of them can be categorized into four classes: synthesis-based model (Tang and Wang, 2003; Lei et al., 2008; Huang and Wang, 2013; Xu et al., 2015), coupled subspace learning (Lin and Tang, 2006; Yi et al., 2015a; Sharma and Jacobs, 2011; Lei et al., 2012), feature representation (Klare et al., 2011; Zhu et al., 2017; Liao et al., 2009; Klare and Jain, 2010) and deep learning methods (Lezama et al., 2017; Wu et al., 2017; Saxena and Verbeek, 2016; He et al., 2017).

Synthesis-based methods (Tang and Wang, 2003; Lei et al., 2008; Huang and Wang, 2013; Xu et al., 2015) learn a mapping from one modality to the other. Once this mapping has been performed, standard homogeneous face recognition approaches can be applied. Tang and Wang (2003) use a multi-scale Markov Random Fields (MRF) model to synthesize sketch drawing from given face photo and vice versa, then apply a Bayesian classifier to distinguish the probing sketch from the synthesized pseudo-sketches. Lei et al. (2008) propose a Canonical Correlation Analysis (CCA) based multi-variant mapping algorithm to reconstruct 3D model from a single 2D near infrared (NIR) image. Huang and Wang (2013) propose a unified model for coupled dictionary and feature space learning which uses a joint dictionary learning to reconstruct face images and then perform face recognition. Xu et al. (2015) learn a dictionary for both visible light (VIS) and NIR domains while forcing the same sparse coefficients for corresponding VIS and

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.cviu.2019.04.003.

* Corresponding author.
E-mail address: huhaif@mail.sysu.edu.cn (H. Hu).

https://doi.org/10.1016/j.cviu.2019.04.003
Received 16 May 2018; Received in revised form 9 April 2019; Accepted 11 April 2019
Available online 18 April 2019

NIR images, so that the coefficients of the NIR image can be used to reconstruct the VIS image and vice versa. However, the synthesis process is actually more difficult than recognition and the performance of these methods heavily depends on the fidelity of the synthesized images (Peng et al., 2015).

Coupled subspace learning methods project both modalities to a common latent space, in which the relevance of data from different modalities can be measured. Lin and Tang (2006) propose Common Discriminant Feature Extraction (CDFE) method to transform data into a common feature space, it takes both inter-modality discriminant information and intra-modality local consistency into consideration. Yi et al. (2015a) use Restricted Boltzmann Machines (RBMs) to learn a shared representation between different modalities, and then apply PCA to remove the redundancy and heterogeneity. Sharma and Jacobs (2011) propose Partial Least Squares (PLS) to linearly map both VIS and NIR domains to a common linear subspace in which they are highly correlated. Lei et al. (2012) develop a Coupled Discriminant Analysis (CDA) based on the locality information in kernel space, where the locality information is incorporated into the CDA as a constraint to improve the generalization ability. The main problem of coupled subspace learning methods is the projection procedure always causes information loss which may decrease the recognition performance (Peng et al., 2015).

Feature representation based methods seek modality-invariant features that are only related to face identity. Klare et al. (2011) present a framework called Local Feature-based Discriminant Analysis (LFDA), where SIFT feature descriptors and Multiscale Local Binary Patterns (MLBP) are used to individually represent both sketches and photos. Multiple discriminant projections are then used on partitioned vectors of the feature-based representation for minimum distance matching. Zhu et al. (2017) develop a novel transductive subspace learning method for heterogeneous face matching, they combine Log-DoG filtering, local encoding and uniform feature normalization together to find better feature representation. Liao et al. (2009) use DoG filtering as preprocessing for illumination normalization, and then employ Multi-block LBP (MB-LBP) to encode NIR as well as VIS images. Klare and Jain (2010) further combine HoG features to LBP descriptors, and utilize sparse representation to improve recognition accuracy. However, most existing methods represent an image ignoring the special spatial structure of faces, which is crucial for HFR in reality (Peng et al., 2015).

Recently, many CNN-based methods are proposed for HFR task, which significantly improve the recognition performance. Lezama et al. (2017) use a CNN that performs a cross-spectral conversion of the NIR image into the VIS spectrum, and introduce Low-rank embedding to restore a low-rank structure for cross-spectral features from the same subject, while enforcing a maximally separated structure for different subjects. Wu et al. (2017) propose a Coupled Deep Learning (CDL) approach by introducing low-rank relevance constraint and cross modal ranking into CNN. Saxena and Verbeek (2016) study different aspects of leveraging a CNN pre-trained on visible spectrum images for heterogeneous face recognition, including extracting features from different CNN layers, fine-tuning the CNN, and using various forms of metric learning. He et al. (2017) develop an effective deep neural network architecture to learn modality invariant representation, where two orthogonal subspaces are embedded to model identity and spectrum information respectively. Wu et al. (2015a,b) develop a lightened CNN network to learn a robust face feature. They introduce a new activation function Max-Feature-Map (MFM) to obtain a compact low-dimensional face representation. And the advantage of the lightened CNN includes small size, fast speed of feature extraction and low-dimensional representation. Though existing deep learning based methods have improved the performance to some extent, HFR task remains a challenging problem and is largely unsolved, which is mainly due to the following two facts. Firstly, HFR data includes two categories of modal information so that it will enlarge the intra-class distance. Therefore, it is difficult for the network to eliminate different modal information and retain the necessary identity information. Secondly, most existing HFR datasets are of small scale (fewer than 10,000 samples) with large feature dimensions (at least 100 × 100 pixels), which may cause over-fitting problem on small-scale training sets. To solve these problems task, in this paper, we present Discriminant Deep Feature Learning Based on Joint Supervision Loss and Multi-layer Feature Fusion (DDFLJM) for HFR. Firstly, instead of using the softmax loss function (Szegedy et al., 2015; Ouyang et al., 2015; Lecun et al., 1998; S and Sun, 2016) as the supervision signal to train the deep model, we proposes a new loss function called Scatter Loss (SL), which embeds both inter- and intra-class information for effectively training the deep model. Secondly, we consider the reuse of mid-level layer features which contain more identity information. Then the SL is applied to multiple layers of network for joint supervision training, which enables multiple layers of the network to obtain discriminative identity features. To eliminate the modality-variant spectrum information, an orthogonality constraint is imposed to mid-level layers of the network for orthogonal decomposition between the modality-invariant identity information and modality-variant spectrum information. Finally, the Modified Gate Two-stream Neural Network (MGTNN) combines multiple-layers features and the Fully Connected (FC) features to form the robust fused features for HFR. Main contributions of our work can be summarized as follows:

- In order to enhance the discriminative power of the deeply learned features, we propose a new supervision signal called Scatter loss which embeds both inter- and intra-class information for effectively training the deep model. The proposed objective function is effective to remove modal information while retaining identity information. The SL is applied to multiple layers of network for joint supervision training, which enables multiple layers of the network to obtain discriminative identity features.
- Dimension Reduction Block (DRB) is designed to extract the auxiliary features on multiple mid-level layers. To reduce spectrum variations of two modalities, an orthogonality constraint is imposed to DRB block for orthogonal decomposition between the modality-invariant identity features and modality-variant spectrum features.
- Due to the pooling layer structure, the high-level layers of the network may lead to the loss of identity information. To make full use of the various layers of the deep network, the multiple-layers features, as a complementary feature of the FC layer, combine the FC features to form the robust fused features through a MGTNN Network.
- Experimental results on the challenging benchmark CASIA NIR-VIS 2.0 (Li et al., 2013) and Oulu-CASIA NIR-VIS (Chen et al., 2009a) databases verify the effectiveness of the proposed model.

2. The proposed DDFLJM model

This section describes the proposed DDFLJM model. As shown in Fig. 1, our model consists of two parts: Feature Extraction Network (FEN) and Feature Fusion Network (FFN). In our model, the proposed SL loss can embed both inter- and intra-class information to make the learned features discriminative. In the FEN network, we adopt the inception-resnet-v1 (Szegedy et al., 2016) as the backbone network. To make full use of various layers of the deep network, DRB block is designed to effectively extract the auxiliary identity features on multiple mid-level layers. An orthogonality constraint is introduced to the DRB block to reduce spectrum variations of two different modalities. In the FFN network, the FC layer features (embeddings) combine the auxiliary features to form the robust fused features through a MGTNN network.

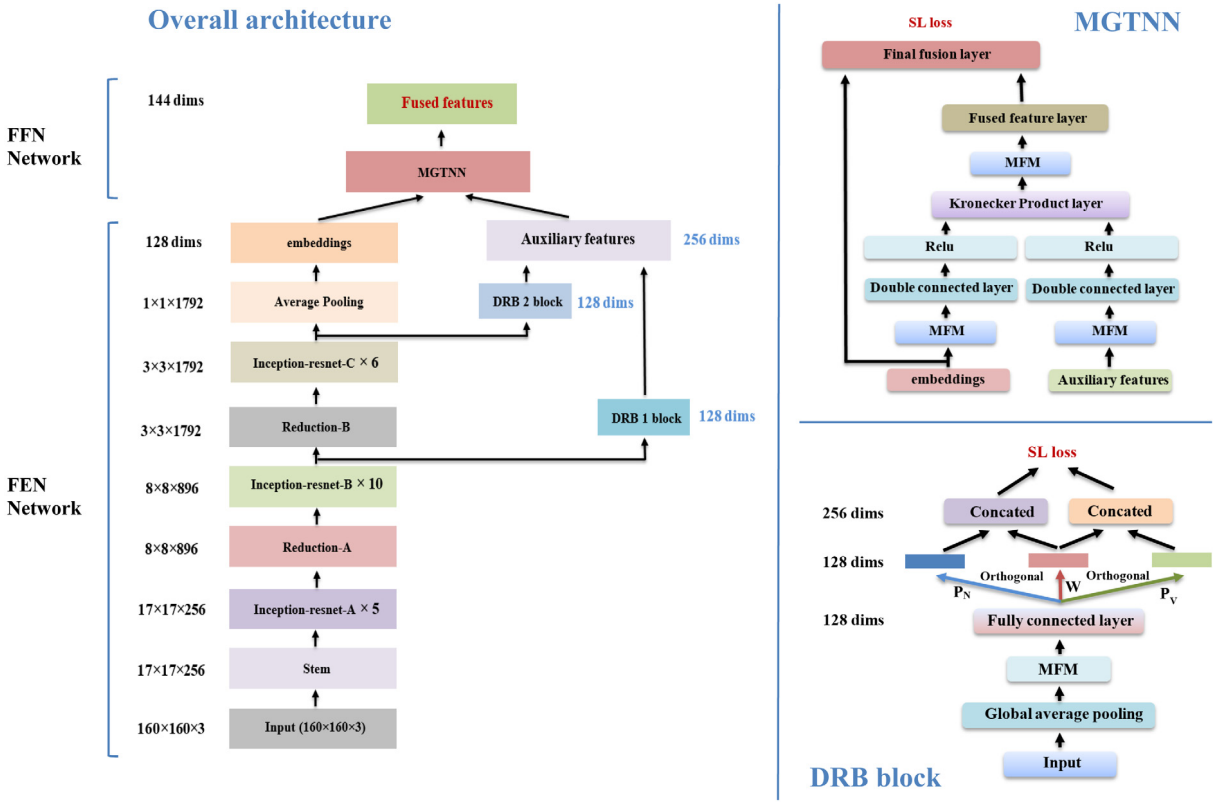

Fig. 1. Overall architecture of the DDFLJM model. Our model consists of two parts: Feature Extraction Network (FEN) and Feature Fusion Network (FFN).

2.1. Network architecture

Table 1 lists the network configurations of DDFLJM, which consists of FEN network and FFN network. In FEN, the inception-resnet-v1 (Szegedy et al., 2016) is adopted as the backbone network, which is a delicately designed and has the merits such as fast speed of feature extraction and low-dimensional representation. The inception-resnet-v1 network consists of five Inception-resnet-A blocks, ten Inception-resnet-B blocks, six Inception-resnet-C blocks, a Reduction-A block, a Reduction-B block and a Stem unit (Szegedy et al., 2016). To make full use of the various layers of the deep network, two DRB blocks are introduced to the mid-level layer of the network. DRB 1 block and DRB 2 block are placed behind the last Inception-resnet-B block and the last Inception-resnet-C block, respectively. As illustrated in Fig. 1, the output of feature maps in the last Inception-resnet-B block are then input to the Reduction-B block and DRB 1 block, respectively. Similarly, the output of feature maps in the last Inception-resnet-C block is fed into Average Pooling layer and DRB 2 block, respectively. The output of the DRB 1 block and DRB 2 block are connected together to form the auxiliary features. In FFN network, the embeddings features combine the auxiliary features to form the robust fused features through the MGTNN network. The details of the MGTNN network and the DRB block refer to upper right and lower right of Fig. 1.

2.2. The proposed SL loss

The deeply learned features need to be discriminative and generalized enough for HFR task. Constructing highly efficient loss function for discriminative feature learning in CNNs is non-trivial. The traditional CNN (Szegedy et al., 2015) employs softmax as the cost function, which does not take inter- and intra-class variations into consideration. Inspired by Fisher criterion (Fisher, 1936), which maximize distance between the classes and minimize distance within the class, in our work, we propose Scatter Loss objective function to map deep features into a discriminate feature space to decrease the intra-class variation while reserving the inter-class variation, which contribute to reducing the gap between different modal domains. For a network with parameter Θ, the proposed Scatter Loss objective function can be expressed as follows:

$$\text{loss}_{SL} = \mathbf{S}_\text{A} + \mathbf{S}_\text{B} \tag{1}$$

where

$$\mathbf{S}_\text{A} = \frac{1}{n}\text{Tr}\left(\sum_{r=1}^{c}\sum_{i=1}^{n_r}\text{I}(l_i = r)(f(x_i;\Theta) - \mathbf{m}_r)(f(x_i;\Theta) - \mathbf{m}_r)^\text{T}\right)$$

$$\mathbf{S}_\text{B} = \frac{1}{c^2 - c}\sum_{i=1}^{c}\sum_{j=1,j\neq i}^{c}\left[\alpha - \text{Tr}\left((\mathbf{m}_i - \mathbf{m}_j)(\mathbf{m}_i - \mathbf{m}_j)^\text{T}\right)\right]_+$$

where $\text{Tr}(\cdot)$ indicates the trace of the matrix, $\text{I}(\cdot)$ is the indicator function with value of 1 or 0, and $f(\cdot)$ is the feature extraction process of the network. $\mathbf{S}_\text{A}$ and $\mathbf{S}_\text{B}$ are intra-class and inter-class distance, respectively. x_i is the ith sample with label l_i, c is the number of categories, and n_r is the number of samples in the rth subject. $\mathbf{m}_r$, $\mathbf{m}_i$ and $\mathbf{m}_j$ are the mean vector of the rth, ith and jth class. α is a margin value between classes.

The proposed SL loss has the following advantages. Firstly, the SL loss minimize distance within the class while maximizing distance between classes, and thus it embeds both intra- and inter-class information to the network for effectively training. Secondly, compared with softmax loss (Szegedy et al., 2015; Deng et al., 2018) which may suffer from massive GPU memory consumption on the classification layer when the identity number increases to million orders of magnitude, the SL can reduce the number of network parameters and the GPU memory consumption. Thirdly, SL can perform well when the samples in each batch are randomly selected, and thus it reduces computation on sample selection. Finally, compared with triplet loss (Schroff et al., 2015), the SL loss is more robust to noise samples and more stable during training process, because the inter-class and intra-class distance are calculated based on the whole class center, which can reduce the effects of noise samples.

Table 1
The network configurations of DDFLJM.

Network or component	Name	Filter size/Stride	Output size	#param
FEN network	Input	–	$160 \times 160 \times 3$	–
	Stem	–	$17 \times 17 \times 256$	692,064
	Inception-resnet-A × 5	–	$17 \times 17 \times 256$	376,320
	Reduction-A	–	$8 \times 8 \times 896$	1,708,032
	Inception-resnet-B × 10	–	$8 \times 8 \times 896$	6,881,280
	Reduction-B	–	$3 \times 3 \times 1792$	3,342,336
	DRB 1 block	–	128	106,496
	Inception-resnet-C × 6	–	$3 \times 3 \times 1792$	9,584,640
	Average pooling	3×3/–	$1 \times 1 \times 1792$	–
	DRB 2 block	–	128	163,840
	embeddings	–	128	229,376
				Total: 23,084,384
FFN network	Input 1 (embeddings features)	–	128	–
	Input 2 (auxiliary features)	–	256	–
	MFM (embeddings)	–	64	–
	MFM (auxiliary)	–	128	–
	Double connected layer (embeddings)	–	16	1024
	Double connected layer (auxiliary)	–	16	2048
	Relu (embeddings)	–	16	–
	Relu (auxiliary)	–	16	–
	Kronecker Product layer	–	256	–
	MFM	–	128	–
	Fused feature layer	–	16	2048
	Final fusion layer	–	144	–
				Total: 4096
DRB 1 block	Input	–	$8 \times 8 \times 896$	–
	Global average pooling	8×8/–	$1 \times 1 \times 896$	–
	MFM	–	448	–
	Fully connected layer	–	128	57,344
	Orthogonal layer $\mathbf{W}$ ($\mathbf{F}_{\mathrm{ID}}$)	–	128	16,384
	Orthogonal layer $\mathbf{P}_{\mathrm{N}}$ (F_{unique})	–	128	16,384
	Orthogonal layer $\mathbf{P}_{\mathrm{V}}$ (F_{unique})	–	128	16,384
				Total: 106,496
DRB 2 block	Input	–	$3 \times 3 \times 1792$	–
	Global average pooling	3×3/–	$1 \times 1 \times 1792$	–
	MFM	–	896	–
	Fully connected layer	–	128	114,688
	Orthogonal layer $\mathbf{W}$ ($\mathbf{F}_{\mathrm{ID}}$)	–	128	16,384
	Orthogonal layer $\mathbf{P}_{\mathrm{N}}$ (F_{unique})	–	128	16,384
	Orthogonal layer $\mathbf{P}_{\mathrm{V}}$ (F_{unique})	–	128	16,384
				Total: 163,840

To optimize the network with SL loss, we adopt the gradient descent following the chain rule. We first calculate the loss value loss_{SL}. Then we propagate the loss to compute the gradient of each layer, and finally we update the parameters according to the gradient. The gradient loss_{SL} w.r.t. $f(x_i;\Theta)$ can be expressed as:

$$\frac{\partial \mathrm{loss}_{SL}}{\partial f(x_i;\Theta)} = \frac{\partial \mathrm{loss}_{SL}}{\partial S_{\mathrm{B}}} \frac{\partial S_{\mathrm{B}}}{\partial f(x_i;\Theta)} + \frac{\partial \mathrm{loss}_{SL}}{\partial S_{\mathrm{A}}} \frac{\partial S_{\mathrm{A}}}{\partial f(x_i;\Theta)} \tag{2}$$

where

$$\frac{\partial \mathrm{loss}_{SL}}{\partial S_{\mathrm{B}}} \frac{\partial S_{\mathrm{B}}}{\partial f(x_i;\Theta)} = \sum_{j=1, j\neq i}^{c} \zeta_j$$

$$\zeta_j = \begin{cases} \frac{2}{n_i} m_j - \frac{2}{n_i} m_i , & \text{if } \mathrm{Tr}\left((\mathbf{m}_i - \mathbf{m}_j)(\mathbf{m}_i - \mathbf{m}_j)^{\mathrm{T}}\right) < \alpha \\ 0 , & \text{otherwise} \end{cases}$$

$$\frac{\partial \mathrm{loss}_{SL}}{\partial S_{\mathrm{A}}} \frac{\partial S_{\mathrm{A}}}{\partial f(x_i;\Theta)} == 2\left(\frac{1}{nn_i} - \frac{1}{n}\right)(f(x_i;\Theta) - m_i)$$

where n_i is the number of samples in the ith class. During the optimization process of network, we adopt data batches manner to update the network. Each mini batch contains several individuals, and each individual contains the same samples.

2.3. Feature extraction network

The feature extraction network consists of a backbone network inception-resnet-v1 and two DRB blocks. The backbone network aims at learning discriminative feature in the FC layer (embeddings layer) while the two DRB blocks are designed to make full use of various layers of the network. In particular, the output of DRB 1 block and DRB 2 block are concatenated together to form the auxiliary features.

2.3.1. Dimension reduction block

In general, the pooling layer of deep network plays the role of feature reduction, which may lead to the loss of useful face information. The deep network (Wu et al., 2015a; Szegedy et al., 2015) only use the FC features (embeddings) without considering other layer features may reduce face recognition performance. Therefore, it is necessary to obtain more useful information for face recognition by making full use of the features of various layer of the network, including the FC layer, the convolution layer and the pooling layer. However, convolutional layer or pooling layer has high feature dimensions and large amount of information redundancy. In order to make use of the features on mid-level layer such as the convolution layer and the pooling layer, as shown in Fig. 1, a DRB block is designed to achieve the goal of dimension reduction and removing identity-unrelated redundant features. Given an intermediate feature map $\mathbf{Q} \in \mathbb{R}^{C\times H\times W}$ as input, the DRB starts with three sequential operations, Global average pooling

(GAP), MFM (Wu et al., 2015a) and Fully Connected layer (FC):

$$\mathbf{k} = f_{\mathrm{FC}}(f_{\mathrm{MFM}}(f_{\mathrm{GAP}}(\mathbf{Q}))) \tag{3}$$

where f_{GAP}, f_{MFM} and f_{FC} denote the operation of GAP, MFM and FC, respectively. $\mathbf{k} \in \mathbb{R}^{1\times d}$ is the output vector of these three sequential operations, and we set d to 128 in our experiment. Inspired by the observation that removing modality-related information is helpful for HFR performance, similar to He et al. (2017, 2018), we further introduce three mapping matrices ($\mathbf{P}_{\mathrm{N}}$, $\mathbf{P}_{\mathrm{V}}$ and $\mathbf{W} \in \mathbb{R}^{d\times p}$) to model variant modality information (such as spectrum information) and identity invariant information. Therefore, the feature representation can be expressed as:

$$\begin{aligned}\mathbf{F}_{\mathrm{N}} &= [\mathbf{F}_{\mathrm{ID}}; \mathbf{F}_{\mathrm{unique}}] = [\mathbf{kW}; \mathbf{kP}_{\mathrm{N}}] \\ \mathbf{F}_{\mathrm{V}} &= [\mathbf{F}_{\mathrm{ID}}; \mathbf{F}_{\mathrm{unique}}] = [\mathbf{kW}; \mathbf{kP}_{\mathrm{V}}]\end{aligned} \tag{4}$$

where $\mathbf{kP}_{\mathrm{N}}$ and $\mathbf{kP}_{\mathrm{V}}$ denotes modality-related feature, $\mathbf{kW}$ denotes identity-related feature, and $\mathbf{F}_i (i \in \{\mathrm{N}, \mathrm{V}\})$ is the final concatenated feature. For better separation of modal information and identity information, an orthogonality constraint is introduced to $\mathbf{W}$ and $\mathbf{P}_i (i \in \{\mathrm{N}, \mathrm{V}\})$ to make them unrelated to each other (He et al., 2017, 2018), i.e.,

$$\begin{aligned}\mathbf{W}^{\mathrm{T}}\mathbf{P}_{\mathrm{N}} &= 0 \\ \mathbf{W}^{\mathrm{T}}\mathbf{P}_{\mathrm{V}} &= 0\end{aligned} \tag{5}$$

The SL loss is used as a supervision signal to enhance the learning feature of DRB, taking the following form,

$$\begin{aligned}&\mathrm{loss}_{DRN} = \mathbf{S}_{\mathrm{A}} + \mathbf{S}_{\mathrm{B}} \\ &\mathrm{s.t.}\ \mathbf{W}^{\mathrm{T}}\mathbf{P}_{\mathrm{N}} = 0 \\ &\mathrm{s.t.}\ \mathbf{W}^{\mathrm{T}}\mathbf{P}_{\mathrm{V}} = 0\end{aligned} \tag{6}$$

where

$$\mathbf{S}_{\mathrm{A}} = \frac{1}{n}\mathrm{Tr}\left(\sum_{r=1}^{\mathrm{c}}\sum_{i\in\{\mathrm{N},\mathrm{V}\}}^{n_r} \mathrm{I}(l_i = r)(\mathbf{F}_i - \mathbf{m}_r)(\mathbf{F}_i - \mathbf{m}_r)^{\mathrm{T}}\right)$$

$$\mathbf{S}_{\mathrm{B}} = \frac{1}{c^2 - c}\sum_{i=1}^{\mathrm{c}}\sum_{j=1, j\neq i}^{\mathrm{c}} \left[\alpha - \mathrm{Tr}\left((\mathbf{m}_i - \mathbf{m}_j)(\mathbf{m}_i - \mathbf{m}_j)^{\mathrm{T}}\right)\right]_+$$

DRB blocks with the supervision signal enable multiple layers of the network to obtain discriminative identity features.

There are three main differences between our orthogonality constraint tricks and those of He et al. (2017, 2018). The first is the design of loss function. He et al. (2017, 2018) adopt two softmax losses as the supervision signal which introduces two classification layers and cannot well solve the over-fitting problem. In our model, we use SL loss to embed both intra- and inter-class information for effectively training. Secondly, He et al. (2017, 2018) introduce the orthogonality constraint tricks on the last layer of the network, while ours introduce the orthogonality constraint trick on the mid-level layer of network. This is because separating spectrum variations in preceding layer of the network may better prevent spectrum-related information from disturbing the embeddings layer. Finally, two DRB blocks with orthogonality constraint are embedded into the designed network, as shown in Fig. 1, which aims at making full use of the features of various layer of the network as well as separating spectrum variations.

As can be seen in Eq. (6), each DRB block has a constrained supervision signal SL, and our goal is to convert them into an unconstrained SL loss. With Lagrange multipliers, (6) can be reformulated as an unconstrained problem:

$$\mathrm{loss}_{DRN} = \lambda_1(\mathbf{S}_{\mathrm{A}} + \mathbf{S}_{\mathrm{B}}) + \lambda_2\left(\left\|\mathbf{W}^{\mathrm{T}}\mathbf{P}_{\mathrm{N}}\right\|_{\mathrm{F}}^2 + \left\|\mathbf{W}^{\mathrm{T}}\mathbf{P}_{\mathrm{V}}\right\|_{\mathrm{F}}^2\right) \tag{7}$$

where λ_1 is weight coefficient of DRB block, and λ_2 is the Lagrange multipliers. Eq. (7) is an unconstrained loss function, so we can use iterative methods to train the network.

2.3.2. Joint supervision loss

The feature extraction network consists of a backbone network inception-resnet-v1 and two DRB blocks. Each DRB block has a constrained supervision signal SL. Therefore, the Joint Supervision Loss for the FFN network contains three items: the SL loss in embeddings layer (loss_{FC}), the constrained SL loss in DRB 1 block (loss_{DRB1}) and the constrained SL loss in DRB 2 block (loss_{DRB2}), taking the following form:

$$\mathrm{loss}_{FEN} = \mathrm{loss}_{FC} + \mathrm{loss}_{DRB1} + \mathrm{loss}_{DRB2} \tag{8}$$

The loss_{FC} (refer to the formula (1)) can enhance the discriminant of the embeddings layer features. While loss_{DRB1} and loss_{DRB2}(refer to the formula (7)) can enhance the discrimination of the auxiliary features. Therefore, the SL is applied to multiple layers of network for joint supervision training, which enables multiple layers of the network to obtain discriminative identity features.

2.4. Feature fusion network

In this subsection, we will show how to fuse the auxiliary features $\mathbf{h}^{(i)} \in \mathbb{R}^{B\times 1}$ and embeddings features $\mathbf{y}^{(i)} \in \mathbb{R}^{D\times 1}$ of the ith image. Although there are a lot of fusion methods, such as concatenation method, these simple fusion methods cannot effectively improve the recognition performance. Based on the GTNN (Hu et al., 2017) method, we propose a Modified GTNN (MGTNN) to achieve the feature fusion, which can fully exploit the complementary information of the embeddings feature and the auxiliary feature, and thus make the fused feature discriminant and robust. As shown on upper right of Fig. 1, the MGTNN is an eight-layer deep network that includes the input layer, the double connected layer, the Relu layer, the Kronecker Product Layer, the Fused feature layer, the Final fusion layer, and two MFM layers. In the input layer, $\mathbf{y}^{(i)}$ and $\mathbf{h}^{(i)}$ are used as inputs for MGTNN, respectively. We employ a MFM (Wu et al., 2015a) activation function to the input feature for dimension reduction, which can simultaneously capture compact representation and competitive information. In the double connected layer of MGTNN, $\mathbf{y}^{(i)}$ and $\mathbf{h}^{(i)}$ are fully connected by their respective parameters in the following form

$$\begin{aligned}\mathbf{F}_y &= \mathbf{U}^{(D)} f_{\mathrm{MFM}}(\mathbf{y}^{(i)}) \\ \mathbf{F}_h &= \mathbf{U}^{(B)} f_{\mathrm{MFM}}(\mathbf{h}^{(i)})\end{aligned} \tag{9}$$

where $\mathbf{F}_y \in \mathbb{R}^{K_D\times 1}$ and $\mathbf{F}_h \in \mathbb{R}^{K_B\times 1}$. $\mathbf{U}^{(D)}$ and $\mathbf{U}^{(B)}$ are size of $K_D \times D/2$ and $K_B \times B/2$ tensors respectively. We then employ a Relu activation function to $\mathbf{F}_y$ and $\mathbf{F}_h$, respectively. In the Kronecker Product layer of MGTNN, $\mathbf{K}_P \in \mathbb{R}^{K_D K_B\times 1}$ is obtained by calculating Kronecker Product of $\mathbf{F}_y$ and $\mathbf{F}_h$ in the following form

$$\mathbf{K}_P = \left(f_{\mathrm{Relu}}(\mathbf{F}_y)\right) \otimes \left(f_{\mathrm{Relu}}(\mathbf{F}_h)\right) \tag{10}$$

where f_{Relu} denotes the Relu operation, and $\otimes$ is the Kronecker Product operation. A MFM activation function is applied to the $\mathbf{K}_P$ for dimension reduction. In the Fused Feature layer of MGTNN, we calculate the fused feature $\mathbf{f}^{(i)} \in \mathbb{R}^{K_C\times 1}$ in the following form

$$\mathbf{f}^{(i)} = \mathbf{S} f_{\mathrm{MFM}}(\mathbf{K}_P) \tag{11}$$

where $\mathbf{S}$ is a size of $K_C \times (K_D K_B/2)$ tensor. In the experiment, we set $K_C = K_D = K_B = K$. In the final fusion layer, the final fusion features $\mathbf{F}_{ff}^{(i)}$ can be obtained by concatenating the embeddings features and the fused feature $\mathbf{f}^{(i)}$, taking the following form:

$$\mathbf{F}_{ff}^{(i)} = \begin{bmatrix}\mathbf{y}^{(i)} \\ \mathbf{f}^{(i)}\end{bmatrix} \tag{12}$$

MGTNN is trained by the SL loss in the following form:

$$\mathrm{loss}_{MGTNN} = \mathbf{S}_{\mathrm{A}} + \mathbf{S}_{\mathrm{B}} \tag{13}$$

where

$$\mathbf{S}_{\mathrm{A}} = \frac{1}{n}\mathrm{Tr}\left(\sum_{r=1}^{\mathrm{c}}\sum_{i=1}^{n_r} \mathrm{I}(l_i = r)(\mathbf{F}_{ff}^{(i)} - \mathbf{m}_r)(\mathbf{F}_{ff}^{(i)} - \mathbf{m}_r)^{\mathrm{T}}\right)$$

$$\mathbf{S}_{\mathrm{B}} = \frac{1}{c^2 - c} \sum_{i=1}^{c} \sum_{j=1, j \neq i}^{c} \left[\alpha - \mathrm{Tr}\left((\mathbf{m}_i - \mathbf{m}_j)(\mathbf{m}_i - \mathbf{m}_j)^{\mathrm{T}}\right)\right]_+$$

There are three main differences between our MGTNN and GTNN. Firstly, we add two MFM activation layer for dimension reduction, because we consider that embeddings features and auxiliary features may contain a lot of redundant information. Secondly, we add a Relu activation layer to increase nonlinearity of feature fusion network. Finally, GTNN adopts the fused feature $\mathbf{f}^{(i)}$ for the final face representation, while MGTNN concatenates the embedding features $\mathbf{y}^{(i)}$ and the fused feature $\mathbf{f}^{(i)}$ to form the final face representation $\mathbf{F}_{ff}^{(i)}$. The designed of MGTNN can retain discriminant embeddings layer features, and obtain complementary information of auxiliary layer through feature fusion network.

2.5. Implementation details

In our works, the open source deep learning framework tensorflow (Abadi et al., 2016) is employed to train the DDFLJM model on TITAN Xp. The training process for DDFLJM consists of two networks: FEN and FFN. MTCNN (Zhang et al., 2016) is used to normalize and crop all training images into 160×160 according to five facial points. In the FEN, we first adopt VIS dataset MS-Celeb-1M (Guo et al., 2016) to pre-train the inception-resnet-v1 with softmax loss. The batch size, the epoch size and training epochs are set to 90, 1000 and 280, respectively. We adopt the dropout ratio of 0.7 for the fully connected layer. The learning rates automatically decrease from 5e−2 to 5e−4. We then adopt the HFR dataset (such as CASIA NIR-VIS 2.0 and Oulu-CASIA NIR-VIS) to finetune the FEN with the Joint Supervision Loss. The batch size and the epoch size are set to 120 (i.e., 20 people per batch, 6 images per person) and 1000, respectively. The training epochs for CASIA NIR-VIS 2.0 and Oulu-CASIA NIR-VIS are set to 18 and 15 in respective. The margin value α, the weight coefficient λ_1 and Lagrange multipliers λ_2 are set to 1.5, 1e−4 and 1e−4. The initial learning rates, learning rate decay epochs and learning rate decay factor are 1e−3, 4 and 0.95, respectively. In particular, we fine tune the FEN by an alternating training manner, that is, we first adopt the loss_{FC} loss to train FEN for one epoch and then adopt $\mathrm{loss}_{DRB1} + \mathrm{loss}_{DRB2}$ loss to train FEN for another epoch. When the FEN training process completed, we start training FFN. We adopt MS-Celeb-1M to train the FFN with loss_{MGTNN} where the size of K is 16. The training epochs, initial learning rates, learning rate decay factor and learning rate decay epochs are 15, 1e−3, 0.95 and 4, respectively.

2.6. Algorithm

The DDFLJM model is outlined in Algorithm 1. The FEN is first pre-trained in MS-Celeb-1M visible face database, and then fine-tuned in HFR dataset. The FFN is trained in MS-Celeb-1M dataset. In the face matching phase, we adopt Joint Bayesian model (Chen et al., 2012) as the classifier which is trained by the fusion features $\mathbf{F}_{ff}^{(i)}$ in HFR dataset.

3. Experiments

In this section, a number of experiments are carried out on two biometric applications in support of the following two objectives:

- Investigate the various properties of the DDFLJM algorithm;
- Evaluate the DDFLJM algorithms on HFR problem by comparing performance with other proposed state-of-the-art methods such as Coupled Simultaneous Local Binary Feature Learning and Encoding (C-SLBFLE) (Lu et al., 2018), Invariant Deep Representation (IDR) (He et al., 2017) and Wasserstein Convolutional Neural Network (WCNN) (He et al., 2018).

Table 2
Recognition performance of different loss on CASIA NIR-VIS 2.0 dataset.

Method	Rank-1 (%)	VR@FAR = 1% (%)	VR@FAR = 0.1% (%)
Softmax loss	84.5 ± 1.7	90.8 ± 1.4	76.2 ± 1.6
Contrastive loss	98.1 ± 0.5	98.9 ± 0.4	94.9 ± 0.9
Triplet loss	97.9 ± 0.5	98.9 ± 0.5	95.6 ± 0.8
SL loss	98.5 ± 0.3	99.2 ± 0.3	97.0 ± 0.5

3.1. Datasets

CASIA NIR-VIS 2.0: CASIA NIR-VIS 2.0 dataset (Li et al., 2013) consists of 725 subjects in total. There are 1–22 VIS and 5–50 NIR face images per subject. Under the View 2 protocol, the evaluation is performed via the tenfold process and in each fold, 357 subjects are used for training while the remaining 358 subjects for testing. Each training fold has approximately 2500 VIS images and 6100 NIR images. In the testing phase, the gallery set has 358 images, that is, each person only contains one VIS image. While the probe set has over 6000 NIR images. According to the protocol (Li et al., 2013), Rank-1 accuracy and Receiver Operating Characteristic (ROC) curve are reported. Fig. 2 shows some samples of cropped VIS-NIR image pairs from the dataset.
Oulu-CASIA NIR-VIS: Oulu-CASIA NIR-VIS dataset (Chen et al., 2009a) is collected with samples only from two views (i.e., VIS and NIR). There are totally 80 subjects with variation of six kinds of expression including happiness, disgust, sadness, anger, fear and surprise. Fifty subjects are selected from Oulu University, while the remaining thirty subjects are selected from CASIA University. According to the protocol in Shao and Fu (2016), forty subjects are selected as a subset, and each subjects contains 48 VIS images and 48 NIR images. In particular, ten subjects are selected from Oulu, and the remaining thirty subjects are from CASIA. The training fold contains twenty subjects with totally 960 VIS images and 960 NIR images. The testing fold also consists of twenty subjects with totally 960 VIS images and 960 NIR images. We report the rank-1 accuracy and the ROC curve according to the protocol. Fig. 3 shows some samples of cropped VIS-NIR image pairs from the dataset.

3.2. Empirical studies of the DDFLJM properties

The following properties of DDFLJM are studied on the CASIA NIR-VIS 2.0 database: The effect of the SL loss, the effects of the margin value α, influence of the number of people and samples in a batch, the effects of the weight coefficient λ_1 and Lagrange multipliers λ_2, the effective of the MGTNN.

***The effect of the SL loss*:** We compare the performance of inception-resnet-v1 network with different loss function such as softmax loss, triplet loss (Schroff et al., 2015), contrastive loss (Sun et al., 2014) and SL loss. Table 2 shows rank-1 accuracy and verification rates of different loss function. In the experiment, we first adopt the MS-Celeb-1M dataset to pre-train the network with softmax loss, and then fine-tune the network with different objective functions in CASIA NIR-VIS 2.0 dataset. Empirically, the margin of the contrastive, triple and SL loss are set to 1.0, 0.2 and 1.5, respectively. The initial learning rates, learning rate decay factor and learning rate decay epochs of the four losses are set to 1e−3, 0.96 and 4, respectively. In softmax loss, the truncated normal initializer with standard deviation 0.1 is used to initialize the classification layer parameters. In SL loss, we randomly select samples and classes (each batch contains 20 people, and each of which has 5 samples). For a fair comparison, the triplet is also randomly selected in triplet loss (each batch contains 90 triplets), and sample pairs are randomly selected in contrastive loss (each batch contains 20 people, each of which has 5 samples, and arbitrary two samples form a sample pairs). We fine-tune the network and stop training until the loss value is stable. In particular, the training epochs of four loss functions (softmax, contrastive, triple and SL loss) are 42, 20, 22 and 18, respectively. The softmax loss has poor performance with rank-1

Algorithm 1 The algorithm for DDFLJM model

Input: Training data $\{\mathbf{x}_i'\}$ in MS-Celeb-1M. Training data $\{\mathbf{x}_i\}$ in HFR dataset. Initialized parameters Θ in FEN. Initialized parameters $\mathbf{U}^{(D)}, \mathbf{U}^{(B)}$ and $\mathbf{S}$ in FFN. Hyper parameter λ_1, λ_2 and α. The number of iteration $t_1, t_2 \leftarrow 0$.

Output: The parameters Θ, $\mathbf{U}^{(D)}, \mathbf{U}^{(B)}$ and $\mathbf{S}$

1: train the inception-resNet-v1 with softmax loss
2: **while** $t_1 <$ iteration number **do**
3: $t_1 \leftarrow t_1 + 1$
3: Compute loss_{FEN} using Eq. 8 in HFR dataset
Update the parameters Θ by gradient descent
4: **end while**
5: **while** $t_2 <$ iteration number **do**
3: $t_2 \leftarrow t_2 + 1$
6: Compute loss_{MGTNN} using Eq. 13 in MS-Celeb-1M
Update the parameters $\mathbf{U}^{(D)}, \mathbf{U}^{(B)}$ and $\mathbf{S}$ by gradient descent
7: **end while**

Fig. 2. Example images of an individual in the CASIA NIR-VIS 2.0 dataset (NIR top, VIS bottom).

Fig. 3. Example images of an individual in the Oulu-CASIA NIR-VIS dataset (NIR top, VIS bottom).

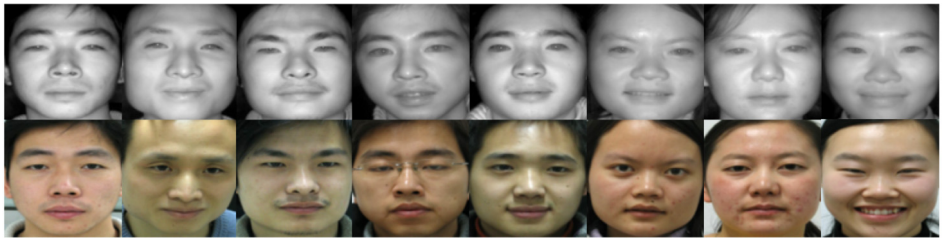

Fig. 4. Example images of the selected eight classes from CASIA NIR-VIS 2.0 dataset (NIR top, VIS bottom).

accuracy of 84.5% and VR@FAR = 0.1% of 76.2%. Because softmax loss introduces more parameters on the classification layer, and it prefers balanced and sufficient training data for each identity. Compared with triplet loss, the contrastive loss has similar performance in rank-1 accuracy and VR@FAR = 1%, but it has poor performance in VR@FAR = 0.1% of 94.9%. It indicates that the triplet loss has better performance than contrastive loss in face verification. The SL loss achieves the best performance in rank-1 accuracy of 98.5% and VR@FAR = 0.1% of 97.0%. In particular, the SL loss is more robust to noise samples and more stable during training process, because the intra-class and inter-class distance are calculated based on the whole class center, which can reduce the effects of noise samples. While contrastive and triplet loss calculate the intra-class and inter-class distance based on samples, and thus they are more sensitive to noise samples. The results indicates that

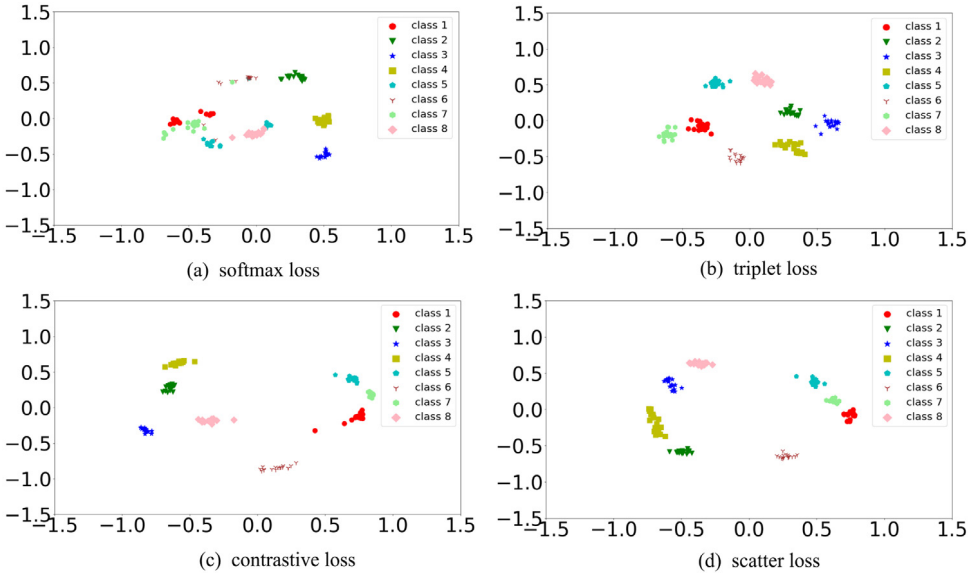

Fig. 5. The distribution of deeply learned features in training set with different losses, (a) softmax loss, (b) triplet loss, (c) contrastive loss, and, (d) scatter loss. The points marked with different colors and markers represent different categories.

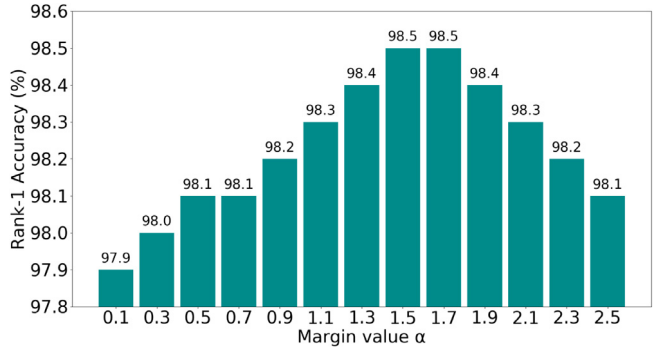

Fig. 6. Recognition performance under different value of α on CASIA NIR-VIS 2.0 dataset.

SL loss can effectively train the network and reserve the discriminative information.

We further investigate the performance of different loss functions in a subset of CASIA NIR-VIS 2.0 dataset. The training set consists of eight subjects randomly selected from CASIA NIR-VIS 2.0, each of which has 19 samples (9 NIR images and 10 VIS images), and example images are shown in Fig. 4. We adopt the inception-resnet-v1 network and train the network with different losses. The embeddings 128-D features conduct dimension reduction through PCA resulting in 2-D deep features. Fig. 5 shows the distribution of deep learning features in the training set. Empirically, the batch size and the number of batches of the four losses are set to 36 and 1000, respectively. The learning rates of softmax, contrastive, triple and SL loss are set to 1e−2, 1e−3, 1e−3 and 1e−3, respectively. The margin of the contrastive, triple and SL loss is set to 0.6. As depicted in the figure, the softmax loss has the worst performance, and there is no clear boundaries between categories. For example, Class 5 and Class 8 have been mixed together. It shows that softmax loss cannot effectively train the network in the NIR-VIS subset. The contrastive, triple and SL loss can effectively train the network in the NIR-VIS subset, and there are clear boundaries between categories. In particular, the SL has better performance in classification, and the intra-class distance is compact, and there are obvious boundaries between classes.

The effects of the margin α: It can be seen from Eq. (1) that the value of the margin α will affect the SL loss performance, and it dominates the inter-class variation. We adopt the inception-resnet-v1 and pre-train it in the MS-Celeb-1M, and then we fine-tune the network with SL loss under different α. The batch size is set to 100 (each batch contains 20 people, and each of which has 5 samples). As depicted in Fig. 6, the network has the lowest rank-1 accuracy of 97.9% when we adopt a small margin $\alpha = 0.1$. When we gradually increase the value of α, the rank-1 accuracy shows a laddering upward trend. When we set α to 1.5 or 1.7, the network achieves the best recognition performance with rank-1 accuracy of 98.5%. The results indicates that small margin between the classes will reduce the inter-class difference and decrease network performance. And a large margin ($\alpha = 1.5$ or 1.7) can increase the distance between classes and improve the network performance. However, when we continue to increase the margin value ($\alpha > 1.7$), the recognition performance shows a stepped downward trend. The results indicates that excessive α makes the network hard to optimize, and thus the network cannot extract discriminative feature. In particular, the rank-1 accuracy is higher than 98.3%, when α is in the appropriate interval ($1.1 \leq \alpha \leq 2.1$). It shows that the learning features are discriminative within a wide range of α.

We further investigate the performance of the SL loss in a subset of CASIA NIR-VIS 2.0 dataset. The subset consists of 8 subjects, each of which has 19 samples (9 NIR images and 10 VIS images), and example images are shown in Fig. 4. We adopt the inception-resnet-v1 network and train the network with SL loss under different margin value α. The embeddings 128-D features conduct dimension reduction through PCA resulting in 2-D deep features. As can be seen in Fig. 7, it shows that the distribution of deeply learned features in training set can be different due to the variant margin α. When we adopt a small margin $\alpha = 0.05$, the inter-class difference becomes small, which may cause misclassification. For example, the classification boundary between class 5 and class 7 is fuzzy, and they have been mixed together. The result indicates that it is difficult to train a network with a small margin (0.05 or 0.3). When we increase the margin ($\alpha = 0.6$ or $\alpha = 1.0$), the inter-class distance becomes large, which is beneficial to classification. Therefore, a small margin between the classes will decrease the network performance, while a large margin will improve inter-class variation and enhance the discriminative power of deep features.

Influence of the number of people and samples in a batch: In the experiment, the inception-resnet-v1 is used for recognition with SL loss on CASIA NIR-VIS 2.0. The margin α is set to 1.5 and samples number is 6 per subject. Fig. 8(a) shows the influence of the number of people

in a batch. The number of people in a batch has slight impact on the rank-1 accuracy. When the number of people changes from 5 to 30 in a batch, the recognition rate is only increased by 0.2. However, the number of people in a batch affects the verification rates. When the number of people in a batch is only 5, the network has lower VR@FAR = 0.1% of 96.0%. When there are more people (>15) in a batch, the SL can effectively train the network and obtain better VR@FAR = 0.1% performance (>96.8%). In particular, the SL achieves best performance on rank-1 accuracy of 98.5% and VR@FAR = 0.1% of 97.0% when the number of people is 20 in a batch. We further investigate the influence of samples per subject. The number of people is fixed to 20 in a batch. As depicted in Fig. 8(b), the number of samples per subject in a batch has slight impact on the rank-1 accuracy and VR@FAR = 0.1%. When the number of samples changes from 2 to 10 in a batch, the rank-1 accuracy varies by less than 0.1 and VR@FAR = 0.1% varies by less than 0.4. The results indicates that the performance of SL loss is minimally affected by the number of people and samples in a batch. Because the inter-class and intra-class distance are calculated based on the whole class center, which can reduce the effects of noise samples.

The effects of the weight coefficient λ_1 and Lagrange multipliers λ_2: It can be seen from Eqs. (7) and (8) that the weight coefficient λ_1 and Lagrange multipliers λ_2 will affect the training process of FEN. In particular, weight coefficient λ_1 is used for balancing loss_{DRB1}, loss_{DRB2} and loss_{FC}. Empirically, we fix the Lagrange multipliers λ_2 to 1e−4. Fig. 9 shows the influence of the weight coefficient λ_1 on the FEN. The discrimination of the embeddings features (backbone network) under different λ_1 is shown in Fig. 9(a), and discrimination of auxiliary features (auxiliary network: DRB 1 and DRB 2) under different λ_1 is shown in Fig. 9(b). The weight coefficient λ_1 has impact on both backbone and auxiliary networks. When we adopt a small weight coefficient λ_1 = 1e−6, the auxiliary network has low rank-1 accuracy of 94.8%. The results indicate that FEN cannot effectively train the auxiliary network when λ_1 has small value, and thus reduces discriminative of the auxiliary features. When we increase the λ_1 to 1e−5, the FEN may pay more attention to the auxiliary network, and the auxiliary network has higher rank-1 accuracy of 95.0%. When we set λ_1 to 1e−4, both the backbone network and auxiliary network achieve the highest rank-1 accuracy of 98.5% and 95.3%, respectively. It indicates that a proper value λ_1 can fully exploit the benefits of the two networks. However, when we continue to increase the weight coefficient (λ_1 > 1e−4), the recognition performance shows a stepped downward trend on both the two networks. Because excessive λ_1 will weakness the effect of backbone network, and thus the FEN network cannot extract discriminative feature. We further investigate the influence of Lagrange multipliers λ_2, and it dominates the effect of orthogonality constraint on DRB 1 block and DRB 2 block. The rank-1 accuracy of backbone network under different λ_2 is shown in Fig. 10(a), and performance of auxiliary network under different λ_2 is shown in Fig. 10(b). Empirically, we fix the weight coefficient λ_1 to 1e−4. Without adopting the orthogonality constraint on the DRB block (i.e., λ_2 = 0), the auxiliary network has rank-1 accuracy of 94.6%. When we adopt the orthogonality constraint and set a small value (λ_2 = 1e−5), the auxiliary network have obvious improvement on rank-1 accuracy of 95.1%. This indicates that orthogonality constraint trick is helpful for DRB block such that it can well separate modality information and identity information. When we set λ_2 to 1e−4, both the backbone network and auxiliary network obtain high rank-1 accuracy of 98.5% and 95.3%, respectively. It indicates that a proper value λ_2 can reduce modality variations of two modalities. However, large value (λ_2 = 0.1) will reduce the performance of backbone network and auxiliary network with rank-1 accuracy of 98.3% and 94.4%, respectively, because large λ_2 will weaken the effect of another target item in Eqs. (7) and (8), and thus FEN cannot effectively extract discriminative feature.

The effective of the MGTNN: As can be seen from Table 3, we evaluate the performance of DDFLJM-MGTNN compared with DDFLJM-embeddings, DDFLJM-auxiliary and DDFLJM-concat. DDFLJM-embeddings, DDFLJM-auxiliary and DDFLJM-MGTNN represent different layer's features of the DDFLJM network (i.e., embeddings layer, auxiliary layer and final face representation $\mathbf{F}_{ff}^{(i)}$). DDFLJM-concat represents that the embeddings features and auxiliary features are fused by concatenation manner. Despite DDFLJM-auxiliary has the worst performance, it still has rank-1 accuracy of 95.3% and VR@FAR = 0.1% of 91.3%, which indicate that the mid-layer of the network contain useful information. The DDFLJM-concat performs worse than DDFLJM-embeddings. It indicates that a simple fusion method cannot improve the recognition performance. DDFLJM-MGTNN achieve the best rank-1 accuracy of 98.8% and VR@FAR = 0.1% of 97.3%, which indicates that mid-level layer contain useful features, and can be used as a complementary feature of the FC layer. The results indicates that different fusion strategies have great influence on the discriminant performance of fused feature. And the designed MGTNN method can fuse the embeddings and auxiliary features to form robust features.

Table 3
The performance of different layer's features of the DDFLJM network on CASIA NIR-VIS 2.0 dataset.

Method	Rank-1 (%)	VR@FAR = 1% (%)	VR@FAR = 0.1% (%)	Dim
DDFLJM-embeddings	98.5 ± 0.3	99.2 ± 0.3	97.1 ± 0.5	128
DDFLJM-auxiliary	95.3 ± 0.5	97.4 ± 0.4	91.3 ± 1.3	256
DDFLJM-concated	97.9 ± 0.4	99.0 ± 0.3	95.6 ± 0.8	384
DDFLJM-MGTNN	98.8 ± 0.3	99.4 ± 0.2	97.3 ± 0.4	144

3.3. Comparison with state-of-the-art methods

In HFR experiments, we evaluate the DDFLJM algorithms on CASIA NIR-VIS 2.0 and Oulu-CASIA NIR-VIS database by comparing performance with other proposed state-of-the-art methods. The parameters setting for the compared algorithms are set according to the published papers (He et al., 2017, 2018).

CASIA NIR-VIS 2.0 dataset: We compare the performance of DDFLJM with some approaches including Common Encoding Feature Discriminant (CEFD) (Gong et al., 2017), Kernel Coupled Spectral Regression (KCSR) (Lei and Li, 2009), Kernel Discriminative Spectral Regression (KDSR) (Huang et al., 2013), Kernel Prototype Similarities (KPS) (Klare and Jain, 2013), H2(LBP3) (Shao and Fu, 2016), Multi-View Discriminant Analysis (MvDA) (Kan et al., 2012), Gabor+RBM (Yi et al., 2015b), C-SLBFLE (Lu et al., 2018), VGG (Parkhi et al., 2015), SeetaFace (Liu et al., 2017), HFR-CNN (Saxena and Verbeek, 2016), HFR-IDNet (Reale et al., 2016), COTS+Low-rank (Lezama et al., 2017), TransfeR NIR-VIS heterogeneous facE recognition neTwork (TRIVET) (Liu et al., 2016), Kernelized Margin-based Cross-Modality Metric Learning (KMCM^{2}L) (Jing et al., 2018), IDR (He et al., 2017) and WCNN (He et al., 2018). Table 4 shows rank-1 accuracy and verification rate of different NIR-VIS face recognition methods, and Fig. 11(a) further plots the ROC curves of the proposed method and its representative competitors. VGG and SeetaFace have poor performance on rank-1 accuracy of 62.1% and 68.0%, respectively. It indicates that the same deep learning approach cannot be simply applied to HFR task due to large domain difference as well as insufficient pairwise images in different modalities during training. Compared with the deep learning method such as IDR, TRIVET and HFR-CNNs et al. the traditional methods H2(LBP3) and MvDA have low rank-1 accuracy and VR@FAR = 0.1%, and their rank-1 accuracy is 43.8% and 41.6%, and VR@FAR = 0.1% is 10.1% and 19.2%, respectively. The results indicate that the deep network can extract discriminative feature for HFR task. Compared to KCSR, KPS, KDSR and MvDA, the traditional method C-SLBFLE achieves better performance on rank-1 accuracy of 86.9% and VR@FAR = 0.1% of 53.0%. Because it divides an input image into several non-overlapped regions and learns the feature mapping and dictionary for each region, which is helpful for reducing modality gap. The KMCM^{2}L method can well solve the HFR problem with rank-1 accuracy of

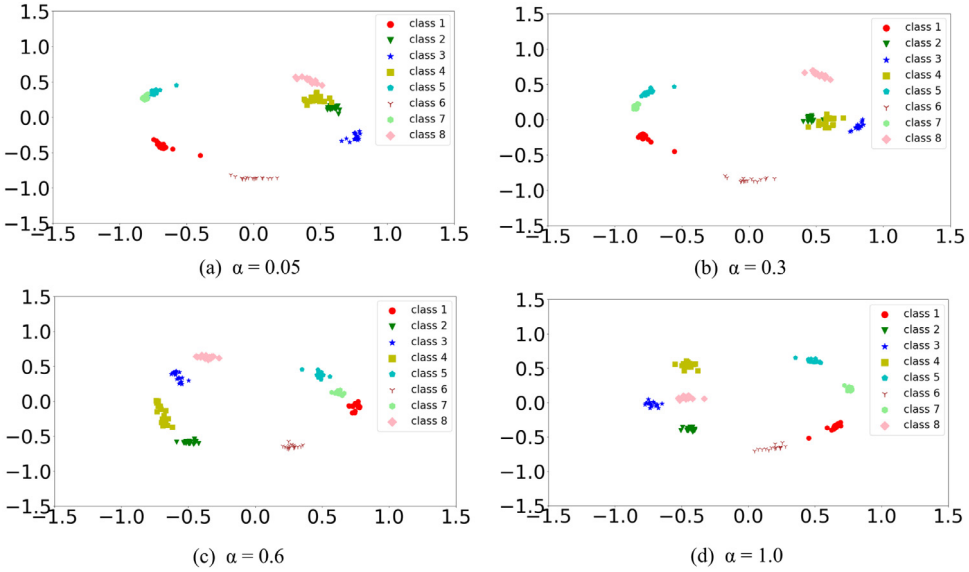

Fig. 7. The distribution of deeply learned features in training set under different margin α, (a) $\alpha = 0.05$, (b) $\alpha = 0.3$, (c) $\alpha = 0.6$, and, (d) $\alpha = 1.0$. The points marked with different colors and markers represent different categories.

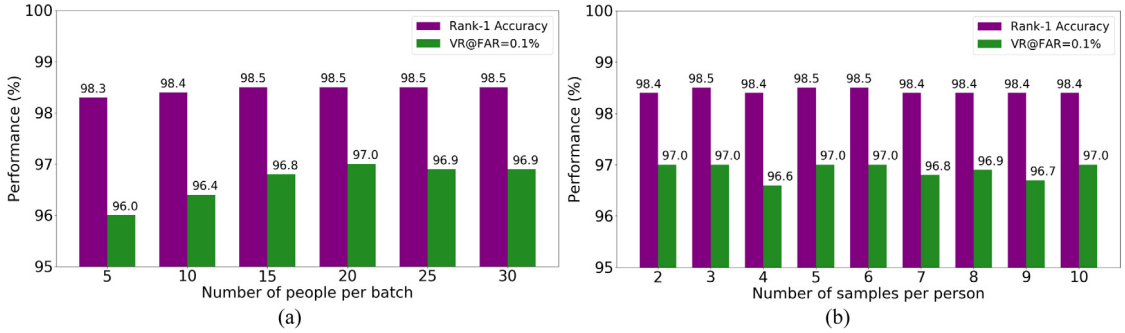

Fig. 8. Influence of the number of people and samples in a batch on CASIA NIR-VIS 2.0, (a) the influence of the number of people in a batch, (b) the influence of the number of samples per subject.

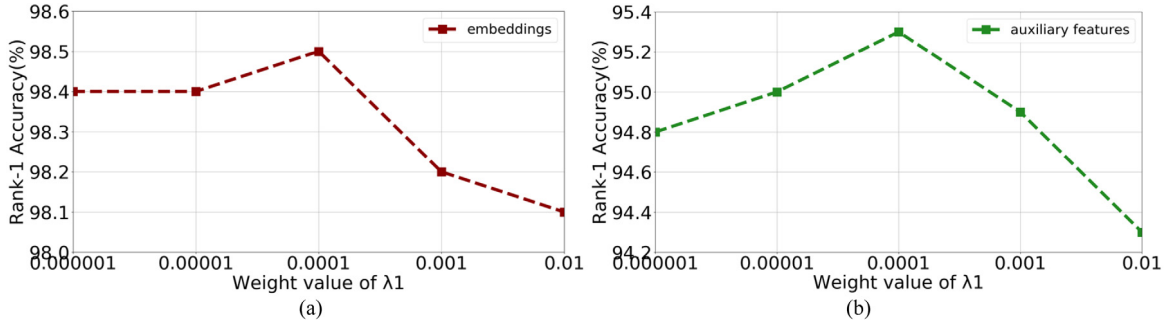

Fig. 9. The influence of the weight coefficient λ_1 on the FEN. (a) The rank-1 accuracy of backbone network under different λ_1. (b) The performance of auxiliary network under different λ_1.

96.5%, which indicates that combining traditional methods with deep learning can improve the performance of HFR. Compared with TRIVET, KMCM^{2}L and IDR, the DDFLJM-MGTNN achieves better performance on rank-1 accuracy of 98.8% and VR@FAR = 0.1% of 97.3%. Because the SL loss embeds both inter- and intra-class information for effectively training the deep model, and an orthogonality constraint is imposed to the DRB block for orthogonal decomposition between the modality-invariant identity features and modality-variant spectrum features. In particular, we combine the embeddings features and auxiliary feature to form the robust fused features through MGTNN network. DDFLJM-MGTNN (98.8%) outperforms WCNN (98.7%) in face recognition, while WCNN outperforms DDFLJM-MGTNN in face verification. Because the WCNN introduces Wasserstein distance to measure the distribution discrepancy between the NIR and the VIS modalities, and a correlation prior is imposed on the fully connected layers of the deep models to alleviate the over-fitting problem on small scale datasets, which can help improve the face verification rates.

Oulu-CASIA NIR-VIS: We compare the performance of DDFLJM with some competitive approaches including KCSR (Lei and Li, 2009),

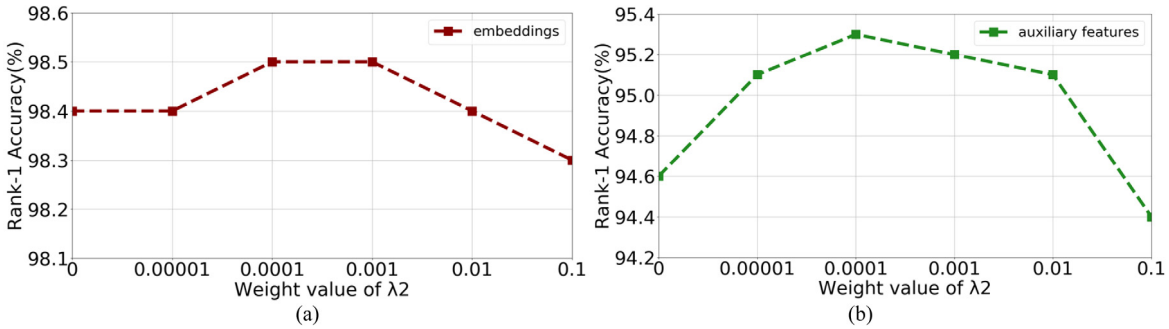

Fig. 10. The influence of the weight coefficient λ_2 on the FEN. (a) The rank-1 accuracy of backbone network under different λ_2. (b) The performance of auxiliary network under different λ_2.

Table 4
Rank-1 accuracy and verification rate on CASIA NIR-VIS 2.0 dataset.

Method	Rank-1 (%)	VR@FAR = 1% (%)	VR@FAR = 0.1% (%)	Dim
KCSR (2009)	33.8	28.5	7.6	–
KPS (2013)	28.2	17.4	3.7	–
KDSR (2013)	37.5	33.0	9.3	–
H2(LBP3) (2016)	43.8	36.5	10.1	–
MvDA (2016)	41.6 ± 4.1	–	19.2	–
Gabor+RBM (2015)	86.2 ± 1.0	–	81.3 ± 1.8	–
CEFD	85.6	–	–	–
C-SLBFLE (2018)	86.9 ± 2.2	80.3	53.0	–
VGG (2015)	62.1 ± 1.9	70.9 ± 1.3	39.7 ± 2.9	4096
SeetaFace (2017)	68.0 ± 1.7	85.2 ± 1.1	58.8 ± 2.3	2048
HFR-CNNs (2016)	85.9 ± 0.9	–	78.0	–
HFR-IDNet (2016)	87.1 ± 0.9	–	74.5	320
COTS+Low-rank (2017)	89.6 ± 0.9	–	–	–
TRIVET (2016)	95.7 ± 0.5	98.1 ± 0.3	91.0 ± 1.3	512
KMCM^{2}L (2018)	96.5 ± 0.4	–	–	–
IDR (2017)	97.3 ± 0.4	98.9 ± 0.3	95.7 ± 0.7	128
WCNN (2018)	98.7 ± 0.3	99.5 ± 0.1	98.4 ± 0.4	128
DDFLJM-embeddings	98.5 ± 0.3	99.2 ± 0.3	97.1 ± 0.5	128
DDFLJM-auxiliary	95.3 ± 0.5	97.4 ± 0.4	91.3 ± 1.3	256
DDFLJM-MGTNN	98.8 ± 0.3	99.4 ± 0.2	97.3 ± 0.4	144

KDSR (Huang et al., 2013), MPL3 (Chen et al., 2009b), H2(LBP3) (Shao and Fu, 2016), KPS (Klare and Jain, 2013), IDR (He et al., 2017), TRIVET (Liu et al., 2016) and WCNN (He et al., 2018). In particular, our method does not compare with approaches such as HFR-CNN, HFR-IDNet, C-SLBFLE and COTS+Low-rank mainly because the results on Oulu-CASIA NIRVIS dataset are not given in the published papers. Table 5 shows the rank-1 accuracy and verification rates of different NIR-VIS methods, and Fig. 11(b) further plot the ROC curves of the proposed method and its representative competitors. The traditional methods KCSR, MPL3 and H2(LBP3) performed poorly in HFR tasks compared to deep learning methods, with rank-1 accuracy of 66.0%, 48.9% and 70.8%, respectively, indicating that traditional methods are difficult to extract robust features. Compared with deep learning-based TRIVET, IDR, and WCNN, our DDFLJM-MGTNN achieves the best performance in rank-1 accuracy and VR@FAR = 0.1%, which is 99.3% and 63.5%, respectively. In particular, the Oulu-CASIA NIR-VIS dataset (1920) contains less training samples than CASIA NIR-VIS 2.0 (about 8600). This finding indicates that the DDFLJM-MGTNN is more suitable for small-scale NIR-VIS HFR dataset than WCNN method, because the WCNN introduces two softmax layers and cannot well solve the over-fitting problem on Oulu-CASIA NIRVIS dataset. The results indicate that DDFLJM-MGTNN method can effectively train the network by using the joint supervision loss function. In addition, to make full use of the various layers of the deep network, the auxiliary features, which effectively extract the features on various layers, combine the embeddings features to form the robust fused features through a MGTNN. Therefore, our DDFLJM-MGTNN method is more suitable for HFR tasks.

Table 5
Rank-1 accuracy and verification rate on Oulu-CASIA NIR-VIS dataset.

Method	Rank-1	VR@FAR = 1% (%)	VR@FAR = 0.1% (%)
MPL3 (2009)	48.9	41.9	11.4
KCSR (2009)	66.0	49.7	26.1
KDSR (2013)	66.9	56.1	31.9
H2(LBP3) (2017)	70.8	62.0	33.6
KPS (2013)	62.2	48.3	22.2
TRIVET (2016)	92.2	67.9	33.6
IDR (2017)	94.3	73.4	46.2
WCNN (2018)	98.0	81.5	54.6
DDFLJM-embeddings	98.5	85.4	60.3
DDFLJM-auxiliary	95.2	75.6	38.1
DDFLJM-MGTNN	**99.3**	**86.1**	**63.5**

3.4. Discussions

We have performed a large number of experiments on HFR task to evaluate our proposed algorithms. From the results presented above, the following observations are made:

- The SL loss achieves better performance compared with softmax loss, contrastive loss, triplet loss on rank-1 accuracy and face verification. Because the inter-class and intra-class distance are calculated based on the whole class center, which can reduce the effects of noise samples and make the training process more stable.
- An orthogonality constraint is imposed to the DRB block for orthogonal decomposition between the modality-invariant identity features and modality-variant spectrum features. The results indicate that orthogonality constraint trick can help improve the HFR performance.
- Compared to the embeddings features, the combination of embeddings and auxiliary features improves Rank-1 accuracy from 98.5% to 98.8% on the CASIA NIR-VIS 2.0 dataset, and the rank-1 accuracy on Oulu-CASIA NIR-VIS increases from 98.5% to 99.3%. It indicates that the reuse of mid-level layer features can further improve the HFR performance. Since the mid-level layer contains useful identity information, how to effectively reuse these features is an undergoing problem.
- The results indicates that the fusion methods have influence on the fused features, and simple concatenation fusion method can even reduce the discrimination of the fused features. In the future, we attempt to design a more effective fusion method that can further improve the HFR performance.
- The DDFLJM approach has three advantages. Firstly, a novel SL loss can enhance the discrimination of learning feature. Secondly, the DRB block with orthogonality constraint can reduce the spectrum variations of two different modalities as well as learning discriminative mid-level features. Thirdly, the designed

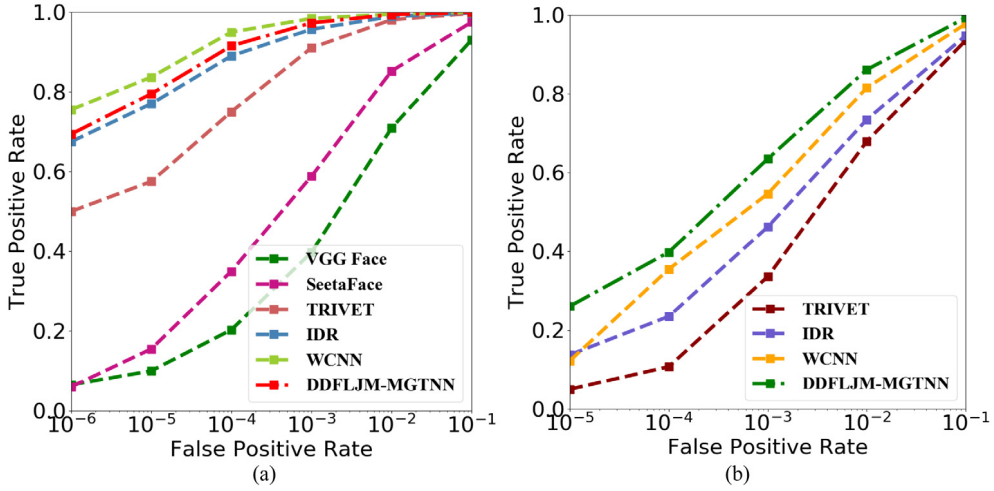

Fig. 11. ROC curves of different methods on the two NIR-VIS datasets, (a) the CASIA NIR-VIS 2.0 database, (b) the Oulu-CASIA NIR-VIS database.

MGTNN can extract robust features and boost the HFR performance. Therefore, the DDFLJM model can be applied to specific tasks of heterogeneous face recognition.

4. Conclusion

In this paper, we have developed a novel Discriminant Deep Feature Learning Based on Joint Supervision Loss and Multi-layer Feature Fusion method for dealing with HFR problem. A novel SL loss which introduces inter-class and intra-class distance based on the whole class center is proposed to effectively train the network and learn discriminative features. An orthogonality constraint is introduced to the DRB block to reduce spectrum variations of two different modalities. Moreover, we combine the FC features and auxiliary features to form the robust fused features through MGTNN fusion method. Experiments on two HFR datasets demonstrate the superiority of our approach.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 61673402, 61273270, and 60802069, in part by the Natural Science Foundation of Guangdong under Grants 2017A030311029, 2016B010109002, in part by the Science and Technology Program of Guangzhou under Grants 201704020180, and in part by the Fundamental Research Funds for the Central Universities of China under Grant 17lgzd08.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467.

Chen, D., Cao, X., Wang, L., Wen, F., Sun, J., 2012. Bayesian Face revisited: a joint formulation. In: Springer European Conference on Computer Vision. pp. 566–579.

Chen, J., Yi, D., Yang, J., Zhao, G., Li, S.Z., Pietikainen, M., 2009a. Learning mappings for face synthesis from near infrared to visual light images. In: IEEE Conf. on Computer Vision and Pattern Recognition. pp. 156–163.

Chen, J., Yi, D., Yang, J., Zhao, G., Li, S.Z., Pietikainen, M., 2009b. Learning mappings for face synthesis from near infrared to visual light images. In: IEEE Conf. on Computer Vision and Pattern Recognition. pp. 156–163.

Deng, J.D., Guo, J., Zafeiriou, S., 2018. Arcface: Additive angular margin loss for deep face recognition. arXiv preprint arXiv:1801.07698.

Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Ann. Human Genet. 7 (2), 179–188.

Gong, D., Li, Z., Huang, W., Li, X., Tao, D., 2017. Heterogeneous face recognition: A common encoding feature discriminant approach. IEEE Trans. Image Process. 26 (5), 2079–2089.

Guo, Y.D., Zhang, L., Hu, Y.X., He, D.X., Gao, J.F., 2016. MS-Celeb-1M: A dataset and benchmark for large-scale face recognition. In: Springer European Conference on Computer Vision. pp. 87–102.

He, R., Wu, X., Sun, Z., Tan, T., 2017. Learning invariant deep representation for NIR-VIS face recognition. In: AAAI Conf. on Artificial Intelligence. pp. 7–18.

He, R., Wu, X., Sun, Z.N., Tan, T.N., 2018. Wasserstein CNN: Learning invariant features for NIR-VIS face recognition. IEEE Trans. Pattern Anal. Mach. Intell. in press.

Hu, G., Hua, Y., Yuan, Y., Zhang, Z., Lu, Z., 2017. Attribute-enhanced face recognition with neural tensor fusion networks. In: IEEE Int. Conf. Computer Vision. pp. 3764–3773.

Huang, X., Lei, Z., Fan, M., Wang, X., Li, S.Z., 2013. Regularized discriminative spectral regression method for heterogeneous face matching. IEEE Trans. Image Process. 22 (1), 353–362.

Huang, D.A., Wang, Y.C.F., 2013. Coupled dictionary and feature space learning with applications to cross-domain image synthesis and recognition. In: IEEE Int. Conf. Computer Vision. pp. 2496–2503.

Jing, H., Yang, G., Shi, Y., et al., 2018. Heterogeneous face recognition by margin-based cross-modality metric learning. IEEE Trans. Cybern. 48 (6), 1814–1826.

Kan, M.N., Shan, S.G., Zhang, H.H., Lao, S.H., Chen, X.L., 2012. Multi-view discriminant analysis. In: Springer European Conference on Computer Vision. pp. 808–821.

Klare, B., Jain, A.K., 2010. Heterogeneous face recognition: Matching NIR to visible light images. In: IEEE Conf. Computer Vision and Pattern Recognition. pp. 1513–1516.

Klare, B.F., Jain, A.K., 2013. Heterogeneous face recognition using kernel prototype similarities. IEEE Trans. Pattern Anal. Mach. Intell. 35 (6), 1410–1422.

Klare, B., Li, Z., Jain, A.K., 2011. Matching forensic sketches to mug shot photos. IEEE Trans. Pattern Anal. Mach. Intell. 33 (3), 639–646.

Lecun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE. 86 (11), 2278–2324.

Lei, Z., Bai, Q., He, R., Li, S.Z., 2008. Face shape recovery from a single image using cca mapping between tensor spaces. In: IEEE Conf. Computer Vision and Pattern Recognition. pp. 1–7.

Lei, Z., Li, S.Z., 2009. Coupled spectral regression for matching heterogeneous faces. In: IEEE Conf. on Computer Vision and Pattern Recognition. pp. 1123–1128.

Lei, Z., Liao, S., Jain, A.K., Li, S.Z., 2012. Coupled discriminant analysis for heterogeneous face recognition. IEEE Trans. Inf. Forensics Secur. 7 (6), 1707–1716.

Lezama, J., Qiu, Q., Sapiro, G., 2017. Not afraid of the dark: NIR-VIS face recognition via cross-spectral hallucination and low-rank embedding. In: IEEE Conf. Computer Vision and Pattern Recognition. pp. 6807–6816.

Li, S.Z., Yi, D., Lei, Z., Liao, S., 2013. The CASIA NIR-VIS 2.0 face database. In: IEEE Conf. Computer Vision and Pattern Recognition Workshops. pp. 348–353.

Liao, S., Yi, D., Lei, Z., Qin, R., Li, S.Z., 2009. Heterogeneous face recognition from local structures of normalized appearance. In: Springer-Verlag Int. Conf. Advances in Biometrics. pp. 209–218.

Lin, D., Tang, X., 2006. Inter-modality face recognition. In: Springer European Conference on Computer Vision. pp. 1–7.

Liu, X., Kan, M.N., Wu, W.L., Shan, S.G., Chen, X.L., 2017. Viplfacenet: an open source deep face recognition SDK. Front. Comput. Sci. 11 (2), 208–218.

Liu, X., Song, L., Wu, X., et al., 2016. Transferring deep representation for NIR-VIS heterogeneous face recognition. In: International Conference on Biometrics. pp. 1–8.

Lu, J.W., Liong, V.E., Zhou, J., 2018. Simultaneous local binary feature learning and encoding for homogeneous and heterogeneous face recognition. IEEE Trans. Pattern Anal. Mach. Intell. 40 (8), 1979–1993.

Ouyang, W., Wang, X., Zeng, X., et al., 2015. Deepid-net: Deformable deep convolutional neural networks for object detection. In: IEEE Conf. Computer Vision and Pattern Recognition. pp. 2403–2412.

Parkhi, O.M., Vedaldi, A., Zisserman, A., 2015. Deep face recognition. In: British Machine Vision Conference. pp. 1–12.

Peng, C., Gao, X., Wang, N., Li, J., 2015. Graphical representation for heterogeneous face recognition. IEEE Trans. Pattern Anal. Mach. Intell. 39 (2), 301–312.

Reale, C., Nasrabadi, N.M., Kwon, H., Chellappa, R., 2016. Seeing the forest from the trees: A holistic approach to near-infrared heterogeneous face recognition. In: IEEE Workshop on Perception beyond the Visible Spectrum. pp. 54–62.

S, He.K.Zhang.X.Ren., Sun, J., 2016. Deep residual learning for image recognition. In: IEEE Conf. Computer Vision and Pattern Recognition. pp. 770–778.

Saxena, S., Verbeek, J., 2016. Heterogeneous face recognition with CNNs. In: European Conference on Computer Vision. Springer International Publishing. pp. 483–491.

Schroff, F., Kalenichenko, D., Philbin, J., 2015. Facenet: A unified embedding for face recognition and clustering. In: IEEE Conf. Computer Vision and Pattern Recognition. pp. 815–823.

Shao, M., Fu, Y., 2016. Cross-modality feature learning through generic hierarchical hyperlingual-words. IEEE Trans. Neural Netw. Learn. Syst. 28 (2), 451–463.

Sharma, A., Jacobs, D.W., 2011. Bypassing synthesis: PLS for face recognition with pose, low-resolution and sketch. In: IEEE Conf. Computer Vision and Pattern Recognition. pp. 593–600.

Sun, Y., Chen, Y., Wang, X., et al., 2014. Deep learning face representation by joint identification-verification. In: Advances in Neural Information Processing Systems. pp. 1988–1996.

Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A., 2016. Inception-v4, inception-resnet and the impact of residual connections on learning. arXiv preprint arXiv:1602.07261.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al., 2015. Going deeper with convolutions. In: IEEE Conf. Computer Vision and Pattern Recognition. pp. 1–9.

Tang, X., Wang, X., 2003. Face sketch synthesis and recognition. In: IEEE Int. Conf. Computer Vision. pp. 687–694.

Wu, X., He, R., Sun, Z., 2015a. A lightened cnn for deep face representation. In: IEEE Conf. Computer Vision and Pattern Recognition. p. 4.

Wu, X., He, R., Sun, Z., Tan, T., 2015b. A light CNN for deep face representation with noisy labels. arXiv preprint arXiv:1511.02683.

Wu, X., Song, L., He, R., Tan, T., 2017. Coupled Deep Learning for Heterogeneous Face Recognition. arXiv preprint arXiv:1704.02450.

Xu, J.F., Pal, D.K., Savvides, M., 2015. NIR-Vis heterogeneous face recognition via cross-spectral joint dictionary learning and reconstruction. In: IEEE Conf. Computer Vision and Pattern Recognition Workshops. pp. 141–150.

Yi, D., Lei, Z., Li, S.Z., 2015a. Shared representation learning for heterogenous face recognition. In: IEEE Int. Conf. and Workshops on Automatic Face and Gesture Recognition. pp. 1–7.

Yi, D., Lei, Z., Li, S.Z., 2015b. Shared representation learning for heterogenous face recognition. In: IEEE Conf. Workshops on Automatic Face and Gesture Recognition. pp. 1–7.

Zhang, K., Zhang, Z., Li, Z., Qiao, Y., 2016. Joint face detection and alignment using multitask Cascaded convolutional networks. IEEE Signal Process. Lett. 23 (10), 1499–1503.

Zhu, J.Y., Zheng, W.S., Lai, J.H., Li, S.Z., 2017. Matching NIR face to VIS face using transduction. IEEE Trans. Inf. Forensics Secur. 9 (3), 501–514.