Kinship Identification through Joint Learning using Kinship Verification Ensembles

Wei Wang, Shaodi You, and Theo Gevers

University of Amsterdam
{w.wang,s.you,th.gevers}@uva.nl

Abstract. Kinship verification is a well-explored task: identifying whether or not two persons are kin. In contrast, kinship identification has been largely ignored so far. Kinship identification aims to further identify the particular type of kinship. An extension to kinship verification run short to properly obtain identification, because existing verification networks are individually trained on specific kinships and do not consider the context between different kinship types. Also, existing kinship verification datasets have biased positive-negative distributions which are different than real-world distributions.
To this end, we propose a novel kinship identification approach based on joint training of kinship verification ensembles and classification modules. We propose to rebalance the training dataset to become more realistic. Large scale experiments demonstrate the appealing performance on kinship identification. The experiments further show significant performance improvement of kinship verification when trained on the same dataset with more realistic distributions.

Keywords: kinship identification, kinship verification ensemble, joint learning

1 Introduction

Kinship is the relationship between people who are biologically related with overlapping genes [17,18], such as parent-children, sibling-sibling, and grandparent-grandchildren [1,20,21,28]. Image-based kinship identification is used in a variety of applications including missing children searching [28], family album organization, forensic investigation [21], automatic image annotation [17], social media analysis [34,6,3], social behavior analysis [14,35,19,11], historical and genealogical research [15,6], and crime scene investigation [16].

While kinship verification is a well-explored task, identifying whether or not persons are kin, kinship identification, which is the task to further identify the particular type of kinship, has been largely ignored so far. Existing kinship verification methods usually train and test each type of kinship model independently [24,20,28] and hence do not fully exploit the complementary information among different kin types. Moreover, existing datasets have unrealistic positive-negative sample distributions. This leads to significant limitations in real world applications. When conducting kinship identification, since there is no prior knowledge
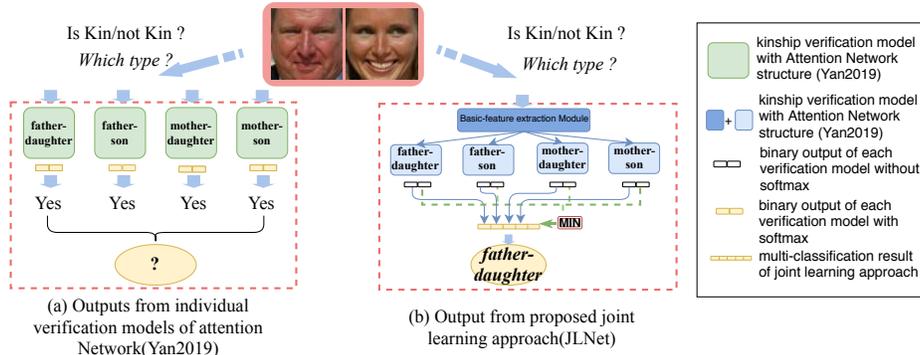
Fig. 1: Identification of kinship relationships using verification ensembles. (a) Existing verification networks are trained independently resulting in contradictory outputs. (b) The output of our proposed joint training

of the distribution of images, all independently trained models are used to determine the kinship type of a specific image pair. Fig. 1 shows an example of providing an image pair to four individually trained verification networks based on a recent state-of-the-art method by Yan *et al.* [33]. The network generates contradictory outputs showing that the test subjects are simultaneously father-daughter, father-son, mother-son and mother-daughter.

In this paper, a new identification method is proposed to learn the identification and verification labels jointly i.e. combining the kinship identification and verification tasks. Specifically, all kinship-type verification models are ensembled by combining the binary output of each verification model to form a multi-class output while training. The binary and multi-class models are leveraged in a multi-task-learning way during the training process to enhance generalization capabilities. Also, we propose a baseline multi-classification neural network for comparison.

We test our proposed kinship identification method on the KinfaceWI and KinfaceWII datasets and demonstrate state-of-the-art performance for kinship identification. We also show that the proposed method significantly improves the performance of kinship verification when trained on the same unbiased dataset.

To summarize, the contributions of our work are:

– We propose a theoretical analysis in metric space of relationships between kinship identification and kinship verification.
– We propose a joint learnt network that simultaneously optimizes the performance of kinship verification and kinship identification.
– The proposed method outperforms existing methods for both kinship identification and unbiased kinship verification.

2   Related Work

Kinship Verification Fang *et al.* [10] are the first to use handcrafted feature descriptors for kinship verification. Later, Xia *et al.* collected a new dataset with young and old parent images to utilize the intermediate distribution using transfer learning [29,30]. Lu *et al.* [18,36] propose a series of metric learning methods. Other handcrafted feature-based methods can be found in [31,27,29,37,7,17,32,9]. Deep learning-based methods [35,33] exploits the advantages of deep feature representations by using pre-trained neural networks in an off-the-shelf way. Zhang *et al.* are the first to use deep convolutional neural networks [35], and Yan *et al.* [33] are the first to add attention mechanisms in deep learning networks for kinship verification. In recent years, there is a trend to combine different features from both traditional descriptors [31,36] and deep neural networks [4,13,22] to generate better representations [2]. (m)DML [8,25] combines auto-encoders with metric learning. However, these methods focus on specific types of kinship and train and test on the same kinship types separately, which may not be feasible in real-world scenarios.

Kinship Identification Different from kinship verification, kinship identification attracted less attention [1]. [1,20] only slightly deal with kinship identification. Guo *et al.* [12] propose a pairwise kinship identification method using a multi-class linear logistic regressor. The method uses graph information from one image with multi inputs. The paper is based on "kinship recognition" and uses a strong assumption that all the data is processed by a perfect kinship verification algorithm. Since there is not sufficient data with family annotations, the method is limited by using multi-input labels. In contrast, our method handles negative pairs and focuses on pair-wise kinship identification. For example, in the context of searching for missing children, we need to handle each potential pair online and find the most likely pair for specific kinship types. In this case, we need to filter the online data and test the most likely data after filtering. As for the family photo arrangement or social media analysis, the aim is to understand the relationships between persons in a picture. There are usually many faces and different kinship relations in a family picture. Hence, the goal is to verify the most likely pairs among negative pairs. Previous methods are not able to cope with this scenario. Fig 2 shows that kinship verification is closely related to kinship identification. As a consequence, we propose a new approach by jointly learning all independent models with kinship verification and identification information.

3   Kinship Identification through Joint Learning with Kinship Verification

In this section, we first introduce the three types of relationship understanding: kinship verification, kinship identification, and kinship classification. Based on this, we introduce the current challenge on kinship identification. Finally, we introduce the concept of conducting kinship identification by using a joint learning strategy between kinship identification and kinship verification.
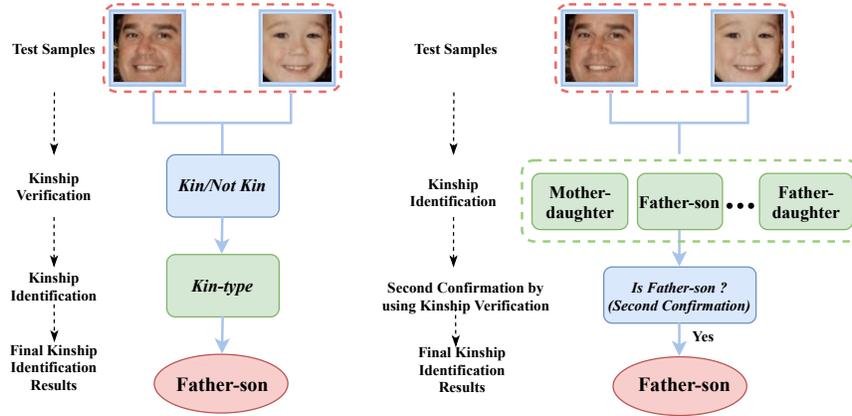
(a) A common process of kinship analysis   (b) Detailed steps of kinship identification

Fig. 2: Flowchart of the relation between kinship verification and kinship identification. (a) Kinship verification is used as a preliminary process for kinship identification. (b) The kinship identification process can be divided into two steps: kinship identification and kinship verification on a specific type.

3.1 Definition of Kinship Verification, Kinship Identification and Kinship Classification

Kinship recognition is the general task of kinship analysis based on visual information. There are mainly three sub-tasks [20,1]: kinship verification, kinship identification, and kinship classification (e.g. family recognition). The goal of kinship verification is to authenticate the relationship between image pairs of persons by determining whether they are blood-related or not. Kinship identification aims at determining the type of kinship relation between persons. Kinship classification [28,20] is the recognition of the family to which a person belongs to. Fig. 2 illustrates the relationship between these tasks. This paper focuses on kinship identification, which is an important but not well-explored topic. Unlike other kinship recognition methods [26,5,23,12], which take images of multiple people as input to predict the relationships between them, the kinship identification task targets at classifying the kin-type of image pairs (negative pairs also included).

3.2 Relationship between Kinship Verification and Kinship Identification and the Limitation of Existing Methods

Relation between the Two Tasks In the literature, kinship verification and identification are two tasks which are studied separately but are closely related. When analyzing the kinship relation between persons, verification is usually applied first to determine whether these persons are kin or not. Then, the kinship type is defined. Fig. 2.a shows the common process of kinship analysis, where

(a) Feature learning phase of different individual trained models for the kinship verification task

(b) Feature learning phase of a multi-classification method for kinship identification task

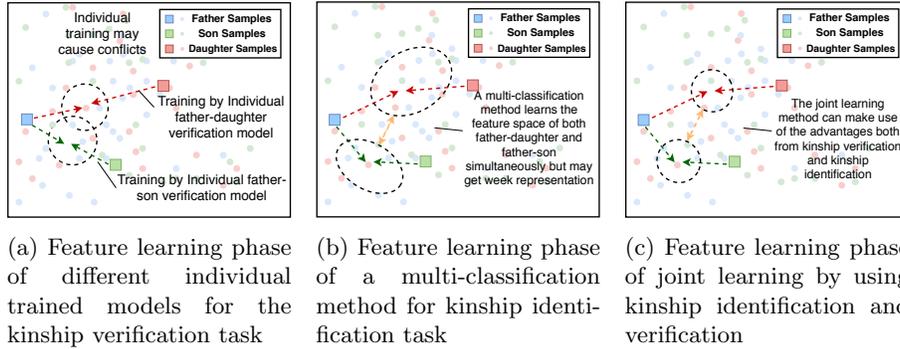(c) Feature learning phase of joint learning by using kinship identification and verification

Fig. 3: Feature space of models during training. Similar feature shapes indicate that the samples are from the same family. Joint learning better represents the context between different kinship relationships. Small circles are used to represent focused samples in feature space.

kinship verification is used as a preliminary process for kinship identification. Furthermore, kinship identification can be divided into two steps, as shown in Fig. 2.b. In the first step, the images are preliminarily classified by the kinship identification model. Then, the classified images are sent to the corresponding verification model. Due to the differences of inherited features among different kin-type images, the kinship verification model provides a better representation than a general kinship identification model. On the other hand, since the kinship identification process filters out irrelevant samples, it provides a consistent and similar feature distribution for kinship verification modelling. In this way, kinship verification and identification are two complementary processes, and can benefit from each other.

Representation of Kinship Relationships in Metric Feature Space and Limitation of Existing Methods In the literature, metric learning is a popular approach for kinship verification. Ideally, the learnt metric space represents kinship likeness for smaller distances. However, existing kinship verification models only consider specific kinship types and ignore the influence of other types.

As shown in Fig. 3a, when the father-daughter verification model is being trained, the features of father and daughter samples will be congregated during the training process and the negative daughter images will be pulled apart. However, due to the negative samples of father-son pairs, which are not included in the training data, the features of son images are less affected by the training process pulling father-son images apart. A narrow-down training of kinship verification can improve the representation of each sample within a specific kin-type. However, since the model does not thoroughly learn other types of negative samples, the separate trained models can easily conflict with each other resulting in ambiguous results. In contrast, a multi-classification method not only

(a) Ensemble Method Based on   (b) Multi-classification Neural Network   (c) Proposed Joint Learning Method
Kinship Verification Models   (Multi-class Net)   (JLNet)
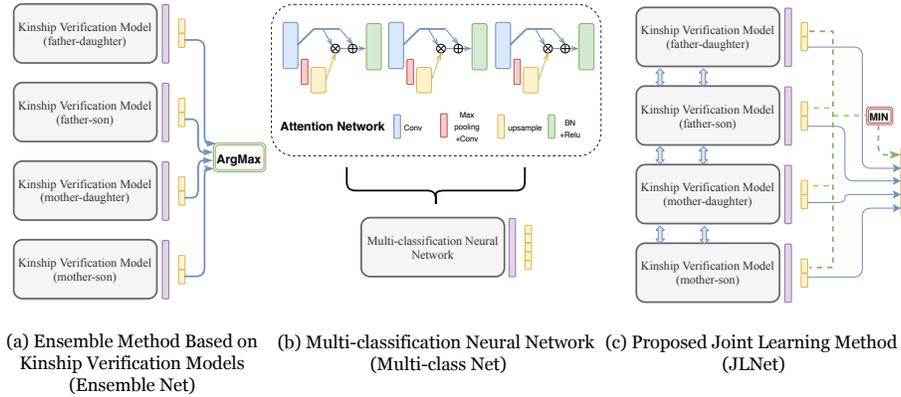(Ensemble Net)

Fig. 4: Structure of the approaches using four relationships as an example.

considers different types of images but also the interaction between different types. As shown in Fig. 3b, the son features will be learned as negative features for the father-daughter feature space, whereas the features of daughters will be considered as negative features for the father-son space. The yellow arrows in Fig. 3b indicate negative samples which will be separated from the matched feature space. A multi-classification method may obtain a weaker representation for a specific kin-type because of the large difference of inherited features among different kin-type images. A joint learning method has the advantage of the generalization of multi-class training and the representation of individual verification models. Hence, identification methods based on joint learning not only repulse negative pairs of different kinship types but also push the potential negative images to the target feature space, which is illustrated in Fig. 3c.

Real World Kinship Distribution and Dataset Bias Note that the proportion of positive and negative samples is highly unbalanced for existing kinship verification datasets. This unbalanced distribution has a negative impact on different applications. Take the online family picture organization application for example. The problem is to determine the matched pairs of images for a specific kinship relationship when the number of kin-related samples only contains a small portion of the entire dataset. Another example is that, when searching for missing children, to retrieve a picture that looks the most like the son of the parents in which the majority of these samples are negative samples.

4   Joint Learning of Kinship Identification and Kinship Verification

We propose a joint learning network (JLNet) based on the learning strategy shown in Fig. 4c aiming to utilize the representation capability of kinship verification models as well as making use of the advantages of multi classification.
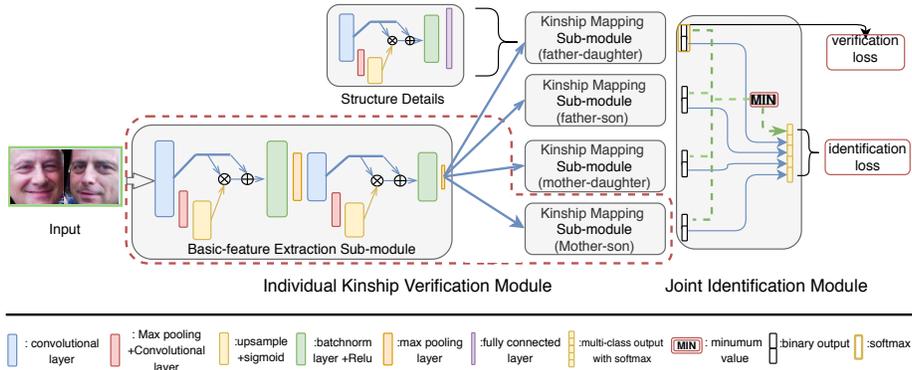
Fig. 5: Architecture of our Joint Learning Network (JLNet)

This approach consists of two major steps: the combination of different types of images and joint learning.

The main ideas of the approach are summarized as follows:

1. We utilize all different kin-types of image pairs to train each kinship model, not based on a specific type.
2. Different models are trained jointly to differentiate negative kinship feature pairs from the matched model and to merge positive pairs as much as possible.

Note that naively using a single classification network (Fig. 4.a) or naively combining multiple verification networks (Fig. 4.b) are not suitable approaches. As described above, our network (Fig. 4.c) utilizes the advantage of both tasks. Without loss of generality, we outline our approach for four relationships: father-daughter (F-D), father-son (F-S), mother-daughter (M-D), mother-son (M-S).

4.1 Architecture of the Proposed Joint Learning Network (JLNet)

The new Joint Learning Network(JLNet) is illustrated in Fig. 5. The structure of JLnet consists of two parts: the individual Verification Module and the Joint Identification Module.

Individual Kinship Verification Module As shown in Fig. 4.c, each Individual Kinship Verification Module is defined as a binary classification problem. Let $S = \left\{ \left(I^\alpha_{p_i}, I^\alpha_{c_i} \right), i = 1, 2, \ldots, N, \alpha = 1, 2, 3, 4, \beta = 1, 2, 3, 4 \right\}$ be the training set of N pairs of images. And $\alpha \in \{1, 2, 3, 4\}$ and $\beta \in \{1, 2, 3, 4\}$ correspond to the following kinship types: father-daughter, father-son, mother-daughter, mother-son respectively. Then, the Individual Verification Module is defined by:

$$\hat{y} = \mathcal{D}^n_\theta \left(I^\alpha_{p_i}, I^\beta_{c_i} \right), \tag{1}$$

where $I_{p_i}^{\alpha} \in \mathbb{R}^{H \times W \times 3}$ ith parent image from α type data set and $I_{c_i}^{\alpha} \in \mathbb{R}^{H \times W \times 3}$ is the jth child image from β type data set. The output $\hat{y}$ of each Individual kinship verification Module is a 1×2 vector. An Attention Network[33] is used as the basic architecture for each Individual Kinship Verification Module. As shown in Fig. 5, the Attention Network uses a bottom-up top-down structure and consists of three attention stages. Each stage consists of one attention module and one residual structure. To exploit the shared information between the complimentary tasks, the parameters of the two stages of the Attention Network are shared to learn low-level and mid-level features from the input images. This forms the Basic-feature Extraction Sub-module. This Basic-feature Extraction Sub-module extracts the basic, generic facial features. Then, high-level features are extracted: four separate branches are added after the last layer (a max pool layer) of the Basic-feature Extraction Sub-module. Each branch focuses on one specific kin-type separately, resulting in four Kinship Mapping Sub-modules. Each of this sub-Module obtains the third stage of the Attention Network and focuses on different kinship types.

Joint Identification Module The binary outputs of each Individual Kinship Verification Module are ensembled. The binary output is described in Eq. 1. The multiple output $\hat{O}$ of the kinship identification module is defined by:

$$\hat{O}_m = \begin{cases} \min_{n \in \{1,2,3,4\}} \mathcal{D}_\theta^n \left(I_{p_i}^{\alpha}, I_{c_j}^{\beta} \right)_{z=1}, & \text{if } m = 0 \\ \mathcal{D}_\theta^m \left(I_{p_i}^{\alpha}, I_{c_j}^{\beta} \right)_{z=2} & \text{if } m \neq 0 \end{cases}, \tag{2}$$

where $m \in \{1,2,3,4,5\}$ represents the mth item of vector $\hat{O}$ and z represents zth item of the output vector of $\mathcal{D}_\theta^n$. The output class C is defined by:

$$C = \underset{z \in \{1,2,3,4,5\}}{\arg\max} \ \sigma(\hat{O})_z, \tag{3}$$

where $\sigma(\cdot)$ is the softmax function.

During the training, the Weighted Cross Entropy loss is used for both kinship verification and identification:

$$\mathcal{L} = -\sum_{i=1}^{n} w_n log(\sigma(\cdot)_n), \tag{4}$$

where n is the class label of the kinship verification or identification output and $\sigma(\cdot)_n$ is the nth output of the softmax function. The loss of the joint learning model is given by a weighted summation of the kinship verification loss (from binary outputs) and the kinship identification loss (from multiple outputs):

$$\mathcal{L} = \sum_{i=1}^{4} \lambda_i \mathcal{L}_{kvi} + \lambda_5 \mathcal{L}_{kI}, \tag{5}$$

where $\mathcal{L}_{kI}$ is the Weighted Cross Entropy loss of the kinship identification output given by Eq. 4 and λ_i is the ith weight of each loss.

4.2   Comparative Methods

Ensemble Method based on Kinship Verification Models (Ensemble Net) Fig. 4.a shows the structure of the Ensemble Method based on Kinship Verification Models (Ensemble Net). The Individual Kinship Verification Modules of the Ensemble Net have the same structure as JLNet. While testing, the Ensemble Net feeds the images into four kinship verification models simultaneously and ensembles four binary outputs. The output class C is defined by:

$$
C = \begin{cases} 0 & \text{if } \max_n \sigma(\mathcal{D}_\theta^n\left(I_{p_i}^\alpha, I_{c_j}^\beta\right))_{z=2} < 0.5 \\ \text{argmax}_n \, \sigma(\mathcal{D}_\theta^n\left(I_{p_i}^\alpha, I_{c_j}^\beta\right))_{z=2}, & \text{otherwise} \end{cases},
$$

$$(6)$$

where $I_{p_i}^\alpha$ is the ith parent image from α type data set and $I_{c_j}^\beta$ is the jth child image from β type data set.

Multi-Classification Neural Network (Multi-class Net) The structure of the Multi-Classification Neural Network (Multi-class Net) is shown in Fig. 4.b. Similar to the Ensemble Net, Multi-class has the same backbone with the Individual Kinship Verification Module of JLNet. The Multi-class Net handles the kinship identification task as a multiple classification problem:

$$
\hat{y} = \mathcal{D}_\theta\left(I_{p_i}^\alpha, I_{c_i}^\beta\right), \tag{7}
$$

where $\boldsymbol{S} = \left\{\left(I_{p_i}^\alpha, I_{c_i}^\beta\right), i = 1, 2, \ldots, N, \alpha = 1, 2, 3, 4, \beta = 1, 2, 3, 4\right\}$ and the output $\hat{y}$ is a 1×5 vector.

5   Experiments

5.1   Unbias Dataset for Training and Testing

Three types of benchmark datasets are generated from the KinfaceWI and KinfaceWII datasets [17,18] consisting of four kinship types: father-daughter (F-D), father-son (F-S), mother-daughter (M-D), mother-son (M-S). To conduct the experiment on unbiased datasets, we re-balance the KinfaceWI and KinfaceWII datasets into three different benchmark datasets as follows:

1. *Independent Kin-type Image Set*: This dataset has four independent subsets, where each subset contains one specific kinship type. This dataset simulates a dataset obtained by an ideal kinship classifier. The split of this image set is the same as KinfaceWI or KinfaceWII. The positive samples are the parent-children pairs with the same type of kinship. The negative samples are the pairs of unrelated parents and children within the same kin-type distribution. The positive and negative ratio is $1 : 1$.

2. *Mixed Kin-Type Image Set*: This dataset combines four different kin-type images taken from the KinfaceWI or KinfaceWII datasets resulting in the type ratio (father-daughter: father-son: mother-daughter: mother-son: negative pairs) to be $1 : 1 : 1 : 1 : 4$. This image set is used for both training and testing. Image pairs with kinship relations are denoted as positive samples. Negative samples are random image pairs without kinship relation but within the same type of distribution.

3. *Real-Scenario Kin-Type Image Set*: This dataset simulates the data distribution for real-world scenarios (e.g. retrieval of missing children). All the images in the kinfaceWI or KinfaceWII datasets are paired one by one, which leads to a highly unbalanced positive-negative rate. Taking KinfaceWII as an example, in each cross-validation, there will be 400 images (200 positive pairs) to be tested. All these images are paired one by one. The ratio of positive and negative pairs is $1 : 398$.

5.2   Experimental Design

All methods are trained on the Mixed Kin-Type Image Set. The dataset is divided into 5-folds and verified by a 5-cross validation. We use the same data augmentation for all methods. The data is augmented by randomly changing the brightness, contrast, and saturation of the image. Random grayscale variations, horizontal flipping, perspective changes, and resizing and cropping are also included. All images have the same size $64 \times 64 \times 3$, and the batch size is set to be 64.

Proposed Joint Learning Method (JLNet) The training scheme of JLNet is divided into two phases. The first one is to train the network parameters for the four models independently. The weighted cross entropy is used for updating and the weight list is set to be $[0.25, 8]$ for each verification output. The second phase is to update network parameters jointly by using both binary and multiple-outputs. The weight matrix of the cross-entropy of the multiple outputs is set to $[0.18, 2, 2, 2, 2]$, and λ_i of the total loss is $1 : 1 : 1 : 1 : 10$ respectively. Adam is used as optimizer and the learning rate is set to 10^{-4}. Since there is no public code available for the attention network, we re-implemented the attention network from scratch. During testing of the kinship verification of each individual kin-type, the binary output of the matched Individual Kinship Verification Module is taken as the final result. During testing of the kinship identification task, both the binary outputs (for kinship verification) and multiple outputs (for kinship identification) are used. A combined result based on the confidence of these two types of outputs are taken as the final result.

Ablation Study

– *Joint Learning without Backpropagation of Multiple Outputs (JLNet†)*: To assess the performance of additional multi-classification outputs, the structure of JLNet† is kept the same as JLNet. Further, JLNet† is trained in the

same way as JLNet, but without using multiple output results for parameter updating.

- *Joint Learning using Multiple Outputs for Kinship Identification (JLNet‡)*: We use the trained model of JLNet directly but only the multiple output is taken as the final result during testing.

Experiments and Comparison

Ensemble Net For Ensemble Net, we provide two ways to train the models:

- *Ensemble Net**: Each verification model is trained separately on the Independent Verification Image Set, which is the same as [33]. This means that each independent kinship verification module is only trained on matched data.
- *Ensemble Net*: Each verification model is trained on the Mixed-Type Image Set, which is the same as the training data of JLNet and Multi-class Net. Adam is used and the learning rate was set to be 10^{-4}. The weights of the cross entropy are $0.25, 8$.

Multi-Class Net Also for the Multi-Class Net, Adam is used as an optimizer. The learning rate is again 10^{-4}. A weight list of $[0.1,1,1,1]$ is used for the weighted Cross Entropy loss.

5.3 Results & Evaluation

The methods are evaluated on the different datasets. Five-cross validation is used as the evaluation protocol. As a reminder, Ensemble Net* is trained on the Independent Kin-Type Kinship Image Set, JLNet† is trained without Back-propagation of Multiple Outputs, and JLNet‡ uses multiple outputs as the final result. The results are shown in Table 1-5.

Table 1: The accuracy of different methods through 5-fold cross-validation on the Independent Kin-Type Image Set.

Methods	KinfaceWI					KinfaceWII				
	F-D	*F-S*	*M-D*	*M-S*	*Mean*	*F-D*	*F-S*	*M-D*	*M-S*	*Mean*
Ensemble Net*	0.7017	0.7506	0.7410	0.615	**0.7021**	0.746	0.7440	0.7520	0.7320	**0.7435**
Multi-class Net	0.6463	0.6797	0.6650	0.5770	0.6420	0.5880	0.6240	0.6200	0.5920	0.6060
Ensemble Net	0.6425	0.6321	0.6382	0.577	0.6224	0.6060	0.6000	0.5860	0.6260	0.6045
JLNet†	0.6534	0.6991	0.6539	0.5772	0.6459	0.6160	0.6100	0.600	0.6500	0.6190
JLNet	0.6608	0.7309	0.7207	0.5897	**0.6755**	0.6800	0.7140	0.6860	0.7060	**0.6965**

Table 2: F1 scores of different methods through 5-fold cross-validation on Independent Kin-Type Image Set

Methods	KinfaceWI					KinfaceWII				
	F-D	F-S	M-D	M-S	Mean	F-D	F-S	M-D	M-S	Mean
Ensemble Net*	0.6915	0.7472	0.7566	0.6648	**0.7150**	0.7671	0.7589	0.7690	0.7607	**0.7639**
Multi-class Net	0.6084	0.6563	0.6767	0.5766	0.6295	0.5629	0.6000	0.6143	0.5062	0.5709
Ensemble Net	0.6639	0.6737	0.6735	0.6083	**0.6548**	0.6213	0.6439	0.6051	0.6399	0.6276
JLNet†	0.6301	0.6952	0.6496	0.5816	0.6391	0.6396	0.6166	0.6061	0.6191	0.6203
JLNet	0.6320	0.7087	0.7052	0.5657	0.6529	0.6585	0.7211	0.6939	0.6847	**0.6896**

Table 3: Macro F1 score and accuracy of kinship identification for the Mixed Kin-Type Kinship Image Set

Methods	KinfaceWI		KinfaceWII	
	macro F1	Accuracy	macro F1	Accuracy
Ensemble Net*	0.3240	0.3723	0.2846	0.3319
Multi-class Net	0.5291	0.5494	0.4861	0.5225
Ensemble Net	0.4837	0.4887	0.4464	0.4564
JLNet†	0.5155	0.5487	0.4648	0.4875
JLNet‡	**0.5507**	0.5880	0.5285	0.5535
JLNet(full)	0.5506	**0.5993**	**0.5343**	**0.5790**

Results for the Independent Kin-Type Image Set Table 1 shows the verification results for the different methods based on the Independent Kin-Type Kinship Image Set. For this image set, accuracy and F1 scores are used to evaluate the performance of kinship verification. All methods are trained on the Mixed Kin-type Image Set except for ensemble Net*. The results show that when trained on the same dataset, JLNet outperforms all other approaches. When tested on the KinfaceWII dataset, JLNet outperforms Multi-Class Net with 9% and Ensemble Net by 9.2% on average accuracy. Considering the F1 score, JLNet outperforms Multi-Class Net with 11.9% and Ensemble Net with 6.2% on average. When comparing JLNet†and JLNet, it is shown that additional multi-outputs improve the results of the ensembled models. When compared with Ensemble Net, the accuracy of JLNet is lower than Ensemble Net. One of the reason is that each of the verification module of Ensemble Net is trained on one specific dataset. This may result in overfitting. JLNet provides better generalization than Ensemble Net*, as shown in the next session.

Results on Mixed Kin-Type Kinship Image Sets Table 3 shows the results of macro F1 scores and accuracy for the kinship identification task using the Mixed Kin-Type Kinship Image Set. The results show that the performances of JLNet outperforms the ensemble and multi-class net methods. Moreover, macro F1 scores show that JLNet(full) outperforms Ensemble Net* with 22.7% on KinfaceWI and with 25.0% on KinfaceWII. Moreover, JLNet(full) outperforms Ensemble Net* with 22.7% on KinfaceWI and 24.7% on KinfaceWII. As shown in

Fig. 6: Confusion matrix for different experiments on the Mixed Kin-Type Image Set using the KinfaceWI dataset. Negative samples are excluded)

Table 4: F10 score and accuracy for different methods on the Real-Scenario Set using KinfaceWI dataset. F10(all) represents the average of F10 scores for all different labels (the negative label is also included)

methods	KinfaceWI						
	F-D	F-S	M-D	M-S	mean	F10(all)	Accuracy
Ensemble Net*	0.0886	0.1179	0.1236	0.1003	0.1076	0.1830	0.4807
Multi-class Net	0.1548	0.2951	0.3047	0.1539	0.2271	0.2947	0.5618
Ensemble Net	0.1508	0.2791	0.2740	0.1378	0.2104	0.2596	0.4537
JLNet†	0.1522	0.2966	0.2937	0.1569	0.2249	0.2985	0.5901
JLNet‡	**0.1742**	0.3235	0.3123	0.1620	0.2430	0.3287	0.6681
JLNet(full)	0.1715	**0.3241**	**0.3198**	**0.1669**	**0.2456**	**0.3459**	**0.7439**

Fig. 6, Ensemble Net* may lead to indecisive results. The independently trained verification models can lead to overfitting and results in weak generalization capabilities. JLNet obtained the highest performance. In Fig. 6, it is shown that the joint learning method provides indecisive results. To this end, the joint learning method JLNet(full) obtains the best performance for kinship identification on the Mixed Kin-type Kinship Image Set.

Results on Real Scenario Sample Set Tables 4 and 5 show the results of the F10 score and accuracy for the kinship identification task in a real-world scenario. We focus more on recall than precision, so the F10 score is used to emphasize on the recall rate. The results show that JLNet(full) obtains the best performance on both KinfaceWI-based Real-Scenario data and KinfaceWII-based Real-Scenario data. The results show that the JLNet(full) outperforms all the other approaches for both KinfaceWI and KinfaceWII. From the confusion matrix in Fig. 7, it is interesting to note that father-son and mother-daughter relations are more distinguishable than other kin-types. We argue that the manifold of pairs with the same gender is easier to be learned.
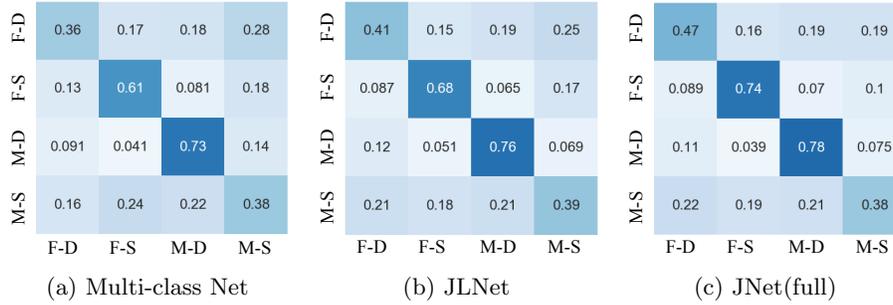
(a) Multi-class Net   (b) JLNet   (c) JNet(full)

Fig. 7: Confusion matrix for different experiments on the Real-Senario Image set using the KinfaceWI dataset. Negative samples are excluded)

Table 5: F10 score and accuracy for different methods on the Real-Scenario Set using KinfaceWII dataset. F10(all) represents the average of F10 scores for all different labels (the negative label is also included)

methods	KinfaceWII						
	F-D	F-S	M-D	M-S	mean	F10(all)	Accuracy
Ensemble Net*	0.0469	0.0713	0.0726	0.0904	0.0703	0.1498	0.4647
Multi-class Net	0.1468	0.1972	0.1853	0.1076	0.1592	0.2528	0.6240
Ensemble Net	0.1399	0.1681	0.1496	0.0900	0.1369	0.2075	0.4874
JLNet†	0.1413	0.1757	0.1624	0.0962	0.1439	0.2303	0.5730
JLNet‡	0.1620	0.2133	0.2127	0.1225	0.1776	0.2735	0.6547
JLNet(full)	**0.1867**	**0.2134**	**0.2296**	**0.1296**	**0.1898**	**0.3003**	**0.7398**

6 Conclusion

In this paper, we presented a new approach for kinship identification by joint learning. Experimental results show that joint learning with kinship verification and identification improves the performance of kinship identification. To our knowledge, this is the first approach to handle the kinship identification tasks by using deep neural networks jointly. Since this method is not restricted to any neural network, a better architecture can further improve the performance for kinship identification.

References

1. Almuashi, M., Hashim, S.Z.M., Mohamad, D., Alkawaz, M.H., Ali, A.: Automated kinship verification and identification through human facial images: a survey. Multimedia Tools and Applications **76**(1), 265–307 (2017)
2. Boutellaa, E., López, M.B., Ait-Aoudia, S., Feng, X., Hadid, A.: Kinship verification from videos using spatio-temporal texture features and deep learning. arXiv preprint arXiv:1708.04069 (2017)
3. Burch, R.L., Gallup Jr, G.G.: Perceptions of paternal resemblance predict family violence. Evolution and Human Behavior **21**(6), 429–435 (2000)

4. Cai, X., Wang, C., Xiao, B., Chen, X., Zhou, J.: Deep nonlinear metric learning with independent subspace analysis for face verification. In: Proceedings of the 20th ACM international conference on Multimedia. pp. 749–752. ACM (2012)
5. Chen, Y.Y., Hsu, W.H., Liao, H.Y.M.: Discovering informative social subgraphs and predicting pairwise relationships from group photos. In: Proceedings of the 20th ACM international conference on Multimedia. pp. 669–678 (2012)
6. DeBruine, L.M., Smith, F.G., Jones, B.C., Roberts, S.C., Petrie, M., Spector, T.D.: Kin recognition signals in adult faces. Vision research **49**(1), 38–43 (2009)
7. Dibeklioglu, H., Ali Salah, A., Gevers, T.: Like father, like son: Facial expression dynamics for kinship verification. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1497–1504 (2013)
8. Ding, Z., Suh, S., Han, J.J., Choi, C., Fu, Y.: Discriminative low-rank metric learning for face recognition. In: 2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG). vol. 1, pp. 1–6. IEEE (2015)
9. Fang, R., Gallagher, A.C., Chen, T., Loui, A.: Kinship classification by modeling facial feature heredity. In: 2013 IEEE International Conference on Image Processing. pp. 2983–2987. IEEE (2013)
10. Fang, R., Tang, K.D., Snavely, N., Chen, T.: Towards computational models of kinship verification. In: 2010 IEEE International conference on image processing. pp. 1577–1580. IEEE (2010)
11. Fessler, D.M., Navarrete, C.D.: Third-party attitudes toward sibling incest: Evidence for westermarck's hypotheses. Evolution and Human Behavior **25**(5), 277–294 (2004)
12. Guo, Y., Dibeklioglu, H., Van der Maaten, L.: Graph-based kinship recognition. In: 2014 22nd international conference on pattern recognition. pp. 4287–4292. IEEE (2014)
13. Huang, G.B., Lee, H., Learned-Miller, E.: Learning hierarchical representations for face verification with convolutional deep belief networks. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 2518–2525. IEEE (2012)
14. Kaminski, G., Dridi, S., Graff, C., Gentaz, E.: Human ability to detect kinship in strangers' faces: effects of the degree of relatedness. Proceedings of the Royal Society B: Biological Sciences **276**(1670), 3193–3200 (2009)
15. KHOURY, M.J., COHEN, B.H., DIAMOND, E.L., CHASE, G.A., MCKUSICK, V.A.: Inbreeding and prereproductive mortality in the old order amish. i. genealogic epidemiology of inbreeding. American journal of epidemiology **125**(3), 453–461 (1987)
16. Kohli, N., Yadav, D., Vatsa, M., Singh, R., Noore, A.: Supervised mixed norm autoencoder for kinship verification in unconstrained videos. IEEE Transactions on Image Processing **28**(3), 1329–1341 (2018)
17. Lu, J., Hu, J., Zhou, X., Shang, Y., Tan, Y.P., Wang, G.: Neighborhood repulsed metric learning for kinship verification. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 2594–2601. IEEE (2012)
18. Lu, J., Zhou, X., Tan, Y.P., Shang, Y., Zhou, J.: Neighborhood repulsed metric learning for kinship verification. IEEE transactions on pattern analysis and machine intelligence **36**(2), 331–345 (2013)
19. Ober, C., Hyslop, T., Hauck, W.W.: Inbreeding effects on fertility in humans: evidence for reproductive compensation. The American Journal of Human Genetics **64**(1), 225–231 (1999)

20. Robinson, J.P., Shao, M., Wu, Y., Fu, Y.: Families in the wild (fiw): Large-scale kinship image database and benchmarks. In: Proceedings of the 24th ACM international conference on Multimedia. pp. 242–246. ACM (2016)
21. Robinson, J.P., Shao, M., Zhao, H., Wu, Y., Gillis, T., Fu, Y.: Recognizing families in the wild (rfiw): Data challenge workshop in conjunction with acm mm 2017. In: Proceedings of the 2017 Workshop on Recognizing Families in the Wild. pp. 5–12. ACM (2017)
22. Sun, Y., Wang, X., Tang, X.: Deep learning face representation from predicting 10,000 classes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1891–1898 (2014)
23. Wang, M., Feng, J., Shu, X., Jie, Z., Tang, J.: Photo to family tree: Deep kinship understanding for nuclear family photos. In: Proceedings of the Joint Workshop of the 4th Workshop on Affective Social Multimedia Computing and first Multi-Modal Affective Computing of Large-Scale Multimedia Data. pp. 41–46 (2018)
24. Wang, S., Ding, Z., Fu, Y.: Cross-generation kinship verification with sparse discriminative metric. IEEE transactions on pattern analysis and machine intelligence **41**(11), 2783–2790 (2018)
25. Wang, S., Robinson, J.P., Fu, Y.: Kinship verification on families in the wild with marginalized denoising metric learning. In: 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017). pp. 216–221. IEEE (2017)
26. Wang, X., Guo, G., Merler, M., Codella, N.C., Rohith, M., Smith, J.R., Kambhamettu, C.: Leveraging multiple cues for recognizing family photos. Image and Vision Computing **58**, 61–75 (2017)
27. Wang, X., Han, T.X., Yan, S.: An hog-lbp human detector with partial occlusion handling. In: 2009 IEEE 12th international conference on computer vision. pp. 32–39. IEEE (2009)
28. Wu, Y., Ding, Z., Liu, H., Robinson, J., Fu, Y.: Kinship classification through latent adaptive subspace. In: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018). pp. 143–149. IEEE (2018)
29. Xia, S., Shao, M., Fu, Y.: Kinship verification through transfer learning. In: Twenty-Second International Joint Conference on Artificial Intelligence (2011)
30. Xia, S., Shao, M., Luo, J., Fu, Y.: Understanding kin relationships in a photo. IEEE Transactions on Multimedia **14**(4), 1046–1056 (2012)
31. Yan, H., Lu, J.: Facial Kinship Verification: A Machine Learning Approach. Springer (2017)
32. Yan, H., Lu, J., Deng, W., Zhou, X.: Discriminative multimetric learning for kinship verification. IEEE Transactions on Information forensics and security **9**(7), 1169–1178 (2014)
33. Yan, H., Wang, S.: Learning part-aware attention networks for kinship verification. Pattern Recognition Letters **128**, 169–175 (2019)
34. Zebrowitz, L.A., Montepare, J.M.: Social psychological face perception: Why appearance matters. Social and personality psychology compass **2**(3), 1497–1517 (2008)
35. Zhang12, K., Huang, Y., Song, C., Wu, H., Wang, L., Intelligence, S.M.: Kinship verification with deep convolutional neural networks (2015)
36. Zhou, X., Hu, J., Lu, J., Shang, Y., Guan, Y.: Kinship verification from facial images under uncontrolled conditions. In: Proceedings of the 19th ACM international conference on Multimedia. pp. 953–956. ACM (2011)

37. Zhou, X., Lu, J., Hu, J., Shang, Y.: Gabor-based gradient orientation pyramid for kinship verification under uncontrolled environments. In: Proceedings of the 20th ACM international conference on Multimedia. pp. 725–728. ACM (2012)