Received: April 9, 2020. Revised: June 20, 2020.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Family-Aware Convolutional Neural Network for Image-based Kinship Verification

Reza Fuad Rachmadi[1]* **I Ketut Eddy Purnama[1]** **Supeno Mardi Susiki Nugroho[1]**
Yoyon Kusnendar Suprapto[1]

[1]*Department of Computer Engineering, Institut Teknologi Sepuluh Nopember, Surabaya 60111, Indonesia*
* Corresponding author's Email: fuad@its.ac.id

Abstract: Faces is a unique region in our body that can be used as a biometric identity. Furthermore, the face between two people that have a kinship relationship may share the same face features which can be used to decide whether two people have a kinship relationship or not. In this paper, we proposed a family-aware convolutional neural network (CNN) for the visual kinship verification problem. Our proposed classifier is constructed by paralleling the state-of-the-art face recognition model and attaching two additional networks, a family-aware network, and a kinship verification network. The family-aware network weights adjusted by learning features specific to the family using deep metric learning loss while the kinship verification network use softmax loss to learn the kinship verification problem. One of the advantages of our proposed classifier is that the output of the classifier is normalized and can be represented as the probability of two images being kin or non-kin. To preserve the face recognition features extraction ability in the state-of-the-art face recognition model, we freeze the weights of the convolutional layers in the classifier for the training process. In the testing process, the family-aware network is detached to construct the final classifier. Experiments on FIW (Families In the Wild) dataset show that our proposed classifier performs better comparing with classifiers that trained without a family-aware network and the ensemble version of the classifier is comparable with several state-of-the-art methods with an average accuracy of 68.84%.

Keywords: Deep metric learning, Family-aware convolutional neural network, Image-based kinship verification.

1 Introduction

In the era big data and internet-of-things, image and video can easily capture using a digital camera, e.g. DLSR Camera or CCTV, and uploaded to social media or photo storage server. One of the most captured objects on the internet is human faces. The human face is a unique region in the human body that can be used as a biometric identity along with fingerprint and retina. Furthermore, human faces can also be used to indicate the kinship relationship (e.g. father-son or mother-daughter) among persons. This is possible due to the descendant of DNA from parents to their children. Two-person that have a kinship relationship may share the same face features (e.g. face shape, eyes, nose, etc.) which can be used to develop a model for kinship verification using face images. To support the development of visual kinship verification, several different visual kinship datasets are formed, including KinFaceW-I [1, 2], KinFaceW-II [1, 2], KFVW (Kinship Face Video in The Wild) [3], Cornell KinFace [4], Tri-Subject Kinship [5] and FIW (Family in The Wild) [6–8]. Although that all of the datasets are potentially used for developing the kinship verification model, Lopez et al. [9] and Dawson et al. [10] proved that not all of the kinship verification dataset is suitable to support the development of kinship verification model.

In this paper, we proposed a family-aware CNN classifier for the visual kinship verification problem. Our proposed classifier constructed by paralleling the state-of-the-art face recognition model (FaceNet [11] and SphereFace [12]) and added additional networks to learn family-aware features. Our contributions can be listed as follows.

DOI: 10.22266/ijies2020.1231.03

- We have investigated the family-aware CNN classifier for the visual kinship verification problem. Experiments on the FIW dataset show that our family-aware features can improve the performance of the CNN classifier comparing with classifier without the additional features.
- We have investigated two different state-of-the-art deep recognition models, FaceNet and SphereFace, to construct our proposed classifier.
- We have investigated two different metric learning loss functions, angular softmax and center loss, that attached at the end of the network to learn family–aware features.

The rest of the paper organized as follows. Section 2 discussed related work on image-based kinship verification problems. The detail of our proposed method is described in section 3. Section 4 and 5 discussed the results of experiments and comparisons with several state-of-the-art methods on image-based kinship verification problems. Lastly, we conclude the experiments in section 6.

2 Related work

In this section, several state-of-the-art methods for the visual kinship verification problem is discussed. We divided this section into two different subsections, metric-based learning and non-metric learning.

2.1 Metric-based learning

Deep metric learning is one of the methods that widely used in the face recognition problem. Some deep metric learning for face recognition tasks that already developed can be found in [11–16]. From those deep metric learning methods, there are several deep metric learning for the visual kinship verification problem that had been investigated by the researcher, including SphereFace [6], and triplet loss [17].

Robinson et al. [6] investigated several different methods that potentially used for the visual kinship verification problem, including non-deep learning methods, metric learning, deep learning, and deep metric learning. The evaluation method was conducted using a 5-fold training/testing split configuration. The experiments show that the best average accuracy achieved by the SphereFace method with an average accuracy of 69.18 ± 3.68.

Li et al. [17] utilize the triplet loss method to training a very deep residual network for the visual kinship verification problem. The classifier is trained using 1 million celeb dataset using softmax loss in the first steps and finetuning the weights on the FIW dataset using triplet loss. In the training process, several data augmentation process was applied to enrich the FIW dataset, including gamma correction, blurring (by up-sampling and down-sampling the input image), and random Gaussian noise. The best accuracy is achieved using an ensemble of ResNet-80, ResNet-101, ResNet-152, and ResNet-269 with an average accuracy of 74.85% using RFIW’17 training/testing split configuration.

2.2 Non-metric learning

Other approaches to solving visual kinship verification problems are by using non-metric learning, including that described by Robinson et al. [6], Dawson et al. [10], and Duan et al. [18].

Robinson et al. [6] investigate several different non-metric learning methods for visual kinship verification problem, including LBP+SVM [19], SIFT+SVM, VGGFace [20], and ResNet-22 [21]. The LBP+SVM and SIFT+SVM method still struggle to verify the kinship relationship with an average accuracy of 55.33 ± 1.01 and 56.80 ± 1.17. The last two deep learning approaches achieve an average accuracy of 61.34 ± 3.81 for ResNet-22 and 64.89 ± 4.68 for VGGFace. Although the average accuracy is lower comparing to state-of-the-art performance, the non-metric learning method is still promising to extend the method and further adjustment or additional method may improve the performance of the classifier.

Dawson et al. [10] found a flaw in several kinship verification datasets. They try to cheat the kinship verification by using FSP (From-Same-Photo) classifier which does not have any knowledge about visual kinship verification. The FSP classifier is used to detect whether two images came from the same photo or not. The training process is done without touching any training data in the kinship verification dataset, instead Dawson et al. creating a new dataset using images taken from the internet to train the classifier. Despite that the FSP classifier does not have any knowledge about face nor visual kinship relationships, the FSP classifier can achieve high accuracy in several datasets. The results indicate that some kinship dataset is not suitable to support the development of kinship verification model due to the high biases that appear in the dataset. Lopez et al. [9] also described that KinFaceW-I and KinFaceW-II are not suitable to support the development of the kinship verification model due to the large bias that appears in the dataset.

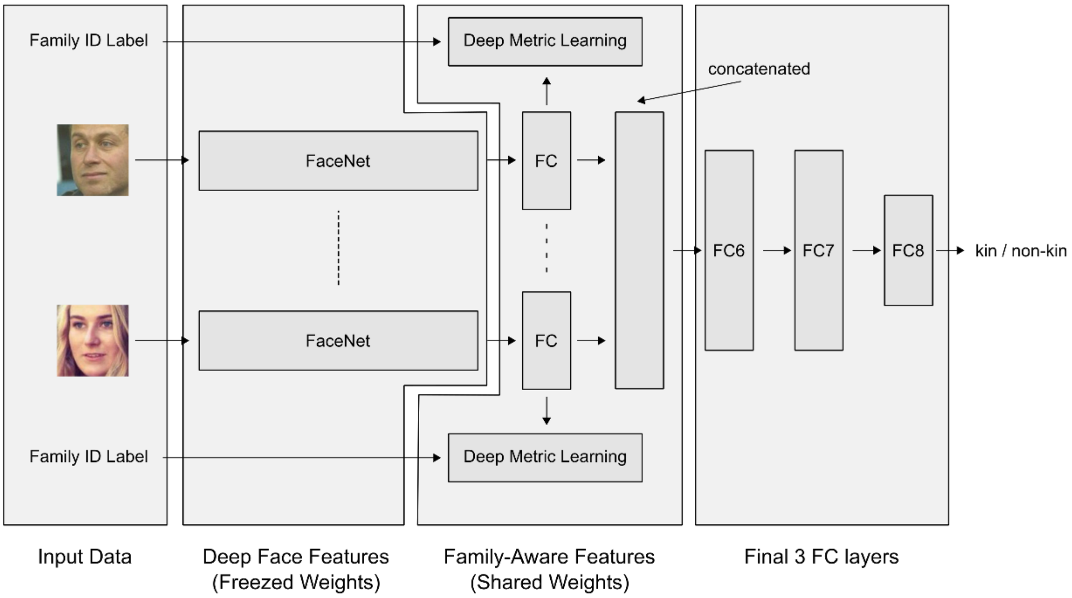

Figure. 1 The diagram of our proposed FA-CNN (Family-Aware Convolutional Neural Network) constructed using facenet

Duan et al. [18] proposed an ensemble of several AdvNet and VGGFace to solve the visual kinship verification problem. The AdvNet constructed from residual CNN architecture by adding the adversarial loss and contrastive loss to learn intermediate features in the classifier. The best average accuracy of 66.58% was achieved using an ensemble of 3 AdvNet and VGGFace classifier. The experiments are done using the FIW dataset with RFIW2017 training/testing split configuration.

2.3 Remarks

As discussed before, we divided the approaches on kinship verification problem into two different categories, metric learning, and non-metric learning solution. The metric learning model is proved very good for the kinship verification problem, but the output of the classifier is a binary label instead of probability which is the downside of the approaches. On the other side, the non-metric learning model produces lower accuracy comparing with the metric learning model but the output of the classifier is a probability that can be interpreted as the confidence of the classifier with the decision. In this paper, we try to combine those two approaches by fed the features learned from deep metric learning to softmax classifier. The combination of those two approaches can improve the performance of the model while preserving the output classifier as a probability of two images being kin or non-kin.

3 Proposed method

The diagram of our proposed FA-CNN classifier constructed based on FaceNet CNN architecture can be viewed in Fig. 1. Our proposed FA-CNN classifier can be also constructed using another state-of-the-art deep face model, such as SphereFace, VGGFace, or ResFace-101.

3.1 Family-aware convolutional neural network

Given a pair of face images, Family-Aware CNN (FA-CNN) classifier will compute the deep features of each face image and classify the pair (kin or non-kin) based on the extracted features. As viewed in Fig. 1, FA-CNN classifier constructed using only the convolutional layers of the state-of-the-art deep face model and attaching an additional network to learn family-aware features and classify the features using three fully-connected layers. The weights in the convolutional are initialized by taking the state-of-the-art deep face model weights and freeze the weights in the training process. We argue that by freezing the weights of convolutional layers, the FA-CNN classifier can preserve the ability to extract discriminate features for face recognition tasks and use those features for visual kinship verification tasks. We freeze the weights based on our preliminary experiments which conclude that freezing the weights can reduce the chance of overfitting and force the classifier to use only face features instead of other features, like FSP (From-Same-Photo) classifier proposed by Dawson et al. [10].

Let f_1 and f_2 is the extracted features using convolutional layers of FA-CNN, the family-aware features are computed using the following equation,

$$\mathrm{fa}_1 = \mathrm{W}^T \mathrm{f}_1 + b \quad (1)$$

$$\mathrm{fa}_1 = \mathrm{W}^T \mathrm{f}_1 + b \quad (2)$$

with W and b is the shared weights and bias for computed the family-aware features. To learn the family-aware features, deep metric learning is used for individual features (f_1 and f_2) in the training process. After the family-aware features extracted, the features than concatenated and goes to three fully-connected layers with two outputs (kin or non-kin) in the last fully-connected layer. The final output of the classifier is computed using softmax function as follows

$$p_c = \frac{exp(\mathrm{w}_c^T \mathrm{x} + b_c)}{\sum_{p=1}^{N} exp(\mathrm{w}_p^T \mathrm{x} + b_p)} \quad (3)$$

where w_k and b_k is the weights and bias used to compute the output of class k. In the training process, the cross-entropy loss function is used to perform weights update in the classifier.

We constructed two different FA-CNN classifiers which can be listed as follows

- FA-CNN with angular loss (FA-AS). In this configuration, the angular loss described in [12] is used to learn the family-aware features.
- FA-CNN with the combination of angular and center loss (FA-ASCL). In this setting, we utilize two metric losses, the angular loss [12] and center loss [28], to learn family-aware features.

4 Deep metric learning

As discussed before, the family-aware features are learned using deep metric learning approaches. There is a lot of deep metric learning that already proposed by researchers and in this subsection, we will focus to discuss two deep metric learning that originally used for face recognition, SphereFace and Center Loss.

4.1 SphereFace

The first deep metric we use to learn family-aware features is SphereFace. SphereFace is the loss function that utilizes the angular or cosines distance. Let $\mathbf{x}$ and W is the input and weights of the last layer of the classifier, the softmax output of the classifier can be computed using equation 3. The angular softmax (A-Softmax) works by transform-ing the features from Euclidean space to angular space. Some other approaches that also use a similar method are CosFace [16], ArcFace [13], and NormFace [15]. In this paper, we use the SphereFace approach which using the following loss function

$$L_a = \frac{1}{N} \sum_{i=1}^{N} -\log\left(\frac{e^{\|\mathbf{x}_i\| \psi(\theta_{c_i,i})}}{e^{\|\mathbf{x}_i\| \psi(\theta_{c_i,i})} + f_s(c_i)}\right) \quad (4)$$

$$f_s(c_i) = \sum_{j \neq c_i} e^{\|\mathbf{x}_i\| \cos(\theta_{j,i})} \quad (5)$$

where $\psi(\theta_{c_i,i})$ define as $\psi(\theta_{c_i,i}) = (-1)^k cos(m\theta_{c_i,i}) - 2k$, $\theta_{c_i,i} \in \left[\frac{k\pi}{m}, \frac{(k+1)\pi}{m}\right]$, and $k \in [0, m-1]$. In the original paper, the m parameter is set to a minimum of $m = 4$ which proved by analysing the lower-bound of m on the binary class case and multi-class case. To conducting the training process for deep metric learning, we use $m = 4$ in our experiments.

4.2 Center loss

Other deep metric learning used in our proposed classifier is center-loss [21]. Unlike SphereFace, center loss is developed based on Euclidean space. The idea of center loss is to minimize the intra-class variation while keeping the features for different classes separable. To minimizing the intra-class variation, the center loss function is used which can be written as follows

$$L_c = \frac{1}{2} \sum_{i=1}^{N} \left\|\mathbf{x}_i - \mathbf{c}_{y_i}\right\|_2^2 \quad (6)$$

where $\mathbf{x}_i$ is the extracted features and $\mathbf{c}_{y_i}$ is the y_i-th class center. Ideally, the class center $\mathbf{c}_{y_i}$ need to be updated when the deep features changed or we can say that to effectively update the class center $\mathbf{c}_{y_i}$, all training data must be taken into account which is impractical for train deep CNN classifier. To solve the problem, Wen et al. [21] proposed a joint loss function between softmax and center loss with additional function to update each class center. The final equation for center loss can be written as follows

$$L = L_s + \lambda L_c$$
$$= -\sum_{i=1}^{N} \log \frac{e^{\mathbf{W}_{y_i}^T \mathbf{x}_i + b_{y_i}}}{\sum_{j=1}^{m} e^{\mathbf{W}_j^T \mathbf{x}_i + b_j}} + \frac{\lambda}{2} \sum_{i=1}^{N} \left\|\mathbf{x}_i - \mathbf{c}_{y_i}\right\|_2^2 \quad (7)$$

where L_s is the softmax loss, L_c is the center loss, λ is a parameter to balancing between intra-class

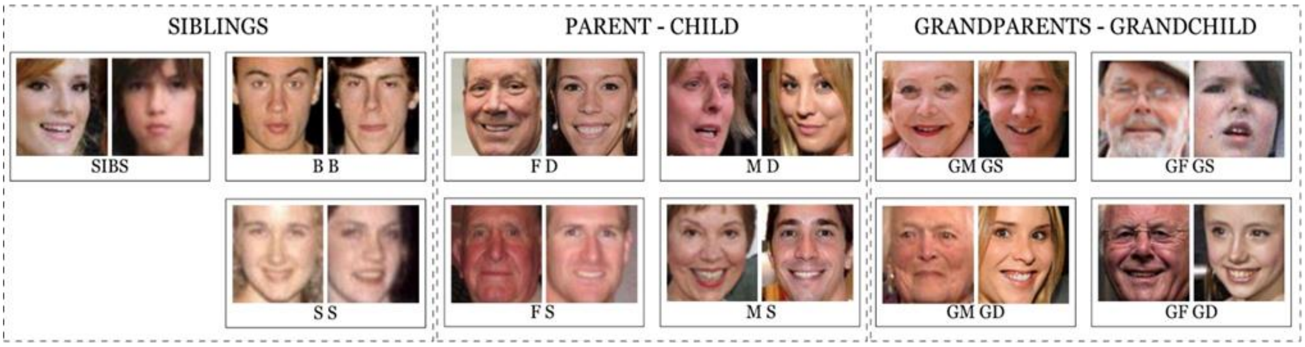

Figure. 2 Examples of face image pair and their kinship relationship on the FIW dataset

variations with separability of the features. The class center $\mathbf{c}_{y_i}$ is updated by taking the gradient of the center loss L_c with respect to $\mathbf{x}_i$ which can be written as follows

$$\frac{\partial L_c}{\partial \mathbf{x}_i} = \mathbf{x}_i - \mathbf{c}_{y_i} \tag{8}$$

$$\Delta\mathbf{c}_j = \frac{\sum_{i=1}^{m} \delta(y_i=j)(\mathbf{c}_j-\mathbf{x}_i)}{1+\sum_{i=1}^{m} \delta(y_i=j)} \tag{9}$$

$$\mathbf{c}_j = \mathbf{c}_j - \alpha\Delta\mathbf{c}_j \tag{10}$$

where α is a parameter to control the update speed of the class center, δ is 1 if the condition in the bracket is true and 0 if the condition in the bracket is false. In our experiments, we set $\lambda = 0.008$ and use the same learning rate α to update the class center $\mathbf{c}_{y_i}$.

5 Results and discussion

In this section, we describe the experiments conducted to evaluate the performance of our proposed family-aware convolutional network. All of the experiments are done using Caffe Deep Learning Framework [22] and FIW dataset [6-8].

5.1 FIW dataset

Family in The Wild (FIW) dataset is a visual kinship verification dataset proposed by Robinson et al. [6-8] in 2016. FIW dataset is the biggest visual kinship verification dataset that currently available. FIW dataset consists of 3 kinship categories, siblings, parent-child, and grandparent-grandchild, with total of 11 kinship relationship class, including father-son (FS), father-daughter (FD), mother-son (MS), mother-daughter (MD), sister (SS), brother (BB), sibling (SIBS), grandfather-grandson (GFGS), grandfather-granddaughter (GFGD), grandmother-grandson (GMGS), and grandmother-granddaughter (GMGD). Although there are several other visual kinship verification datasets, we use the FIW dataset as our main evaluation dataset based on the results discussed in [10] and [9]. The FIW dataset consists of 11,932 natural family photos of 1,000 family and 656,954 face pairs for 11 different kinship relationship types. Fig. 2 shows face pairs for 11 kinship relationships on the FIW dataset.

One of the disadvantages when using the FIW dataset for evaluating a proposed method is that there is a lot of variant for training/testing split configuration. The FIW dataset has a least 4 different training and testing split configuration, including 5-fold configuration, RFIW (Recognize Family in The Wild) 2016 challenge configuration, RFIW 2017 configuration, RFIW 2018 configuration, and RFIW 2020 configuration. Those variants of training/testing split configuration will limit the comparison between proposed methods.

In this paper, we use the original 5-fold training and testing split configuration in the experiments. The 5-fold training/testing split configuration is very challenging configuration due to the no overlapping family appears between the fold. To perform the training process, we change the visual kinship verification problem to binary classification (the pair have kinship relationship or not) instead of 11 different binary classification problems.

5.2 Training dan testing process

The training process is done by constructed the FA-CNN classifier as showed in Fig. 1 and freeze the state-of-the-art face recognition model weights. The weights that freeze will preserve the ability of the classifier to extract the face features for face recognition tasks which based on our experiments will reduce the overfitting problem in the training process.

The training process is done for around 16-24 epochs (depending on the fold split configuration) and NAG (Nesterov Accelerated Gradient) [23] training algorithm. The learning rate is set to 0.001 and reduce by polynomial policy so that in the last iteration the learning rate will be 0. We only use

Table 1. Verification results (%) on FIW dataset for 5-fold experiments.

No.	Method	siblings			parent-child				grandparent-grandchild				Avg
		SS	BB	SIBS	FD	FS	MD	MS	GFGD	GFGS	GMGD	GMGS	
1	P-FaceNet	72.75	66.48	68.54	66.47	66.40	68.66	67.88	62.38	60.59	60.23	60.34	65.52
2	P-FaceNet + FA-AS-SC1	75.42	69.30	71.32	68.41	68.09	71.15	70.52	62.32	61.38	62.16	61.88	67.46
3	P-FaceNet + FA-AS-SC2	76.44	70.59	71.85	68.88	68.04	71.24	70.53	61.74	60.70	62.74	62.85	67.78
4	P-FaceNet + FA-ASCL-SC1	75.93	70.14	71.39	69.29	68.34	71.57	70.55	62.38	61.68	63.29	62.74	67.94
5	P-FaceNet + FA-ASCL-SC2	76.52	69.98	71.40	68.41	67.24	70.87	69.89	61.09	60.35	62.18	62.46	67.31
6	P-SphereFace	70.28	65.01	66.00	65.89	65.84	67.54	65.92	60.75	60.07	59.78	59.90	64.27
7	P-SphereFace + FA-AS-SC1	73.39	67.15	69.39	67.87	67.78	69.84	68.70	64.02	61.50	62.59	61.60	66.72
8	P-SphereFace + FA-AS-SC2	74.57	68.25	69.82	67.63	68.20	70.38	70.19	61.57	61.73	61.13	61.79	66.85
9	P-SphereFace + FA-ASCL-SC1	73.62	68.05	69.65	67.37	67.28	70.12	68.73	63.09	61.47	62.21	62.09	66.70
10	P-SphereFace + FA-ASCL-SC2	74.36	69.25	69.84	66.72	67.22	70.34	68.82	61.24	61.05	61.18	61.58	66.51

simple data augmentation (random crop) by resizing the input image to 256×256 and cropping the image randomly to 224×224. The family-aware features will be learn using three different configurations, angular softmax loss or sphereface loss, center loss, and the combination between those two losses. For sphereface loss, we use an annealing optimization strategy which also used in the original SphereFace paper [12]. Let the original softmax loss function $L = \frac{1}{N}\sum_i -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right)$, the annealing optimization strategy works by changing the function f_{y_i} with $f_{y_i} = \frac{\lambda\|\mathbf{x_i}\|\cos\left(\theta_{y_i}\right) + \|\mathbf{x_i}\|\psi\left(\theta_{y_i}\right)}{\lambda+1}$, such that the equation consists of original softmax loss function and angular softmax loss function with λ as a hyperparameter to control the proportion of the original softmax loss function, During the training process, we start set large λ as initialization and gradually reduce λ during training. We conducted the training process using two different scenarios which described as follows

- **Scenario 1 (SC1)**. We initialize the $\lambda = 1000$ and reduce to $\lambda = 150$ for 2 epoch, $\lambda = 50$ for 6 epoch respectively, and perform the final finetuning for 8 epoch by freeze the deep metric layer. At the first 8 epoch we use sigmoid cross-entropy loss for the kinship verification layer and changing the loss function in the final finetuning to softmax loss function.
- **Scenario 2 (SC2)**. In this scenario we want to further reduce the λ to 15. We initialize $\lambda = 1000$, reduce to $\lambda = 150$ for 2 epoch, $\lambda = 50$ for 2 epoch, $\lambda = 25$ for 4 epoch, $\lambda = 15$ for 8 epoch respectively, and perform the final finetuning for 8 epoch. Same as used in scenario 1, we utilize the sigmoid cross-entropy loss function while reducing the λ and change it to softmax loss function in the final finetuning.

In the testing process, the angular softmax loss layer is detached and only the family-aware features used for the testing process. The input image in the testing process is resized to 256×256 and cropping into 10 different crop regions (left-top, left-bottom, right-top, and right-bottom) with a resolution of 224×224. The final classification decision is taken by averaging the prediction scores from those 10 different crop regions. To decide whether a pair have a kinship relationship or not, we use the threshold of 0.3 which 0.2 lower comparing with the standard threshold. The threshold is decided by performing a validation process using validation data in the training process.

5.3 Results

The summary of the FA-CNN experiments using the FIW dataset with 5-fold training/testing split configuration can be viewed in Table 1. We evaluate two different base CNN architecture that is taken

Table 2. Verification results (%) on FIW dataset for 5-fold experiment using ensemble configuration. The number in the ensemble configuration is associated with single classifier in Table 1

No.	Method	siblings			parent-child				grandparent-grandchild				Avg
		SS	BB	SIBS	FD	FS	MD	MS	GFGD	GFGS	GMGD	GMGS	
1	Ensemble (2) + (3)	76.50	70.39	72.22	69.22	68.43	71.68	71.07	62.11	61.11	62.30	62.50	67.96
2	Ensemble (4) + (5)	76.93	70.63	72.02	69.44	68.18	71.91	70.89	62.33	60.83	62.75	62.27	68.02
3	Ensemble (2) - (5)	77.30	70.79	72.60	69.94	68.76	72.36	71.66	62.40	61.58	63.10	62.43	68.45
4	Ensemble (7) + (8)	74.64	68.04	70.23	68.28	68.50	70.98	69.40	63.09	62.32	62.37	62.18	67.28
5	Ensemble (9) + (10)	74.40	69.05	70.33	67.43	67.60	70.79	69.29	62.62	61.35	61.82	62.19	66.99
6	Ensemble (7) - (10)	75.28	68.95	70.80	68.53	68.58	71.29	69.82	63.60	61.57	61.91	62.73	67.55
7	Ensemble (2) – (5) & (7) – (10)	77.32	70.93	72.67	70.13	69.59	72.89	71.90	64.19	61.68	62.94	62.91	68.84

from the deep face recognition model, including SphereFace and FaceNet. The weights for SphereFace CNN architecture are taken from the original implementation of SphereFace while the weights for FaceNet CNN architecture are taken from trained the FaceNet CNN classifier on LFW using center loss. As viewed in Table 1, the best average accuracy is achieved using the P-FaceNet+FA-ASCL-SC1 classifier with an average accuracy of 67.94%. The grandparent-grandchild kinship relationship is the most difficult case for our proposed classifier with a maximum average accuracy of around 62%. Our proposed classifier is struggling to classify grandparent-grandchild kinship relationship types due to the limited data available in the training dataset. The simplification process of the problem (from individual kinship relationship to kin/non-kin classification problem) may also be contributed to the performance of the classifier on grandparent-grandchild kinship relationship types. The best performance of our proposed classifier is achieved on sister-sister (SS) kinship relationship type with an average accuracy of around 74-77%. Comparing with the classifier without family-aware features, our proposed can improve the performance of the classifier by around 2%.

To further improve the performance of our proposed classifier, we conducted additional experiments by ensemble several classifiers. Ensemble configuration is constructed by taking several single classifiers that achieve top performance as showed in Table 1. Table 2 shows a summary of our experiments using the ensemble classifier on the FIW dataset. By using ensemble configuration, the performance of our proposed classifier is increased around 1-2%. The best performance of the ensemble classifier is achieved using an ensemble of top 8 single classifiers with an average accuracy of 68.84%. As showed in Table 2, all kinship relationship types performance is increased except for grandparent-grandchild types which achieved in around the same accuracy as the non-ensemble classifier.

5.4 Experiment on RFIW 2017

To provide more comprehensive performance evaluation, we tested our proposed model using FIW dataset with training/testing split configuration used for RFIW 2017 challenge. Unlike the dataset used in the previous experiment, the FIW dataset with RFIW 2017 split configuration only consists of two kinship relationship types, siblings and parent-child.

The training process is done using the same scenario and hyperparameters as used in the 5-fold RFIW dataset. Table 3 shows the validation results on the FIW dataset using RFIW 2017 split configuration. As showed in Table 3, the best accuracy is achieved using the ensemble of four P- FaceNet classifiers that trained using two different scenarios with an average accuracy of 72.39%. Unfortunately, our proposed classifier cannot be tested using the testing dataset on RFIW 2017 because the submission server is closed by the organizer.

Table 3. Validation results (%) on FIW dataset using RFIW 2017 split configuration

No.	Method	siblings			parent-child				Avg
		SS	BB	SIBS	FD	FS	MD	MS	
1	P-FaceNet	71.26	66.1	70.55	66.51	67.14	68.59	68.02	68.31
2	P-FaceNet + FA-AS-SC1	75.05	71.5	71.15	68.9	70.17	71.2	71.12	71.29
3	P-FaceNet + FA-AS-SC2	75.86	71.61	72.06	66.67	68.87	69.55	70.83	70.78
4	P-FaceNet + FA-ASCL-SC1	75.39	73.31	72.12	69.32	70.99	71.84	71.13	72.01
5	P-FaceNet + FA-ASCL-SC2	76.13	70.77	71.68	66.31	68.65	70.49	69.77	70.54
6	Ensemble (2) + (3)	76.47	72.91	72.27	68.38	70.37	71.27	72.11	71.96
7	Ensemble (4) + (5)	76.45	73.03	71.99	68.19	70.33	71.46	70.67	71.73
8	Ensemble (2) – (5)	76.72	73.73	73.11	68.69	70.92	71.79	71.77	72.39

Table 4. Validation results (%) on FIW dataset using RFIW 2018 split configuration

No.	Method	siblings			parent-child				grandparent-grandchild				Avg
		SS	BB	SIBS	FD	FS	MD	MS	GFGD	GFGS	GMGD	GMGS	
1	P-FaceNet	68.86	74.40	73.67	66.69	67.33	67.17	68.35	56.14	60.85	56.97	58.94	65.39
2	P-FaceNet + FA-AS-SC1	70.65	75.46	77.52	68.63	68.91	68.93	70.56	56.66	63.13	59.02	58.94	67.12
3	P-FaceNet + FA-AS-SC2	72.77	76.16	78.06	69.41	68.73	70.40	70.61	56.23	63.55	59.04	60.08	67.73
4	P-FaceNet + FA-ASCL-SC1	71.98	75.53	76.9	69.10	69.37	69.87	70.68	58.25	65.16	60.77	59.76	67.94
5	P-FaceNet + FA-ASCL-SC2	73.58	76.44	77.55	69.21	68.54	69.99	69.84	56.76	65.16	59.55	59.97	67.87
6	Ensemble (2) + (3)	71.98	75.96	78.08	69.35	69.11	70.01	70.98	56.35	64.43	59.18	59.05	67.68
7	Ensemble (4) + (5)	73.2	76.07	77.71	69.71	69.26	70.3	70.69	57.86	65.52	60.96	59.49	68.25
8	Ensemble (2) – (5)	72.95	76.19	78.41	69.55	69.46	70.66	71.22	57.05	64.69	60.48	59.59	68.20

5.5 Experiment on RFIW 2018

Additionally, we also tested our proposed model using RFIW 2018 dataset. The training process is done for around 20 epoch (we reduce the epoch after analyzing the validation results) and the same hyperparameter configuration as used in the previous experiment. Unlike the RFIW 2017 dataset that only consists of two type kinship relationships, the grandparent-grandchild kinship relationship is added in the RFIW 2018 dataset.

Table 4 shows the validation results on RFIW 2018 validation set. As showed in Table 4, the best accuracy is achieved using an ensemble of P-FaceNet + FA-ASCL-SC1 and P-FaceNet + FA-ASCL-SC2 with an average accuracy of 68.25 %. Same as the experiments on the FIW dataset using 5-fold training/testing split configuration, the proposed classifiers also struggle to classify grandparent-grandchild kinship types. To compare the performance of our proposed classifier with other approaches, we tested our best model with the RFIW 2018 testing set and achieved the 2nd best performance with an average accuracy of 66.96 %.

5.6 Time Execution

To analyze the characteristic of our proposed classifier, we compute the number of parameters and execution time of our proposed classifier. For GPU, we tested our proposed classifier on NVIDIA TITAN

Table 5. The number of parameters and execution time of our proposed classifier.

No.	Classifier	#Params	Execution Time (ms)	
			GPU	CPU
1	P-FaceNet + FA- AS	73.94 M	16.82	1498
2	P-FaceNet + FA- ASCL	102.26 M	17.52	1536
3	P-SphereFace + FA- AS	77.84 M	11.38	947
4	P-SphereFace + FA- ASCL	131.32 M	12.05	998

Table 6. Comparison with several state-of-the-art classifier on FIW dataset.

No.	Method	Split	Avg. Acc
1	SphereFace [6]	5-fold	69.18 %
2	SDMLoss [24]	5-Fold	68.68 %
3	LASL [25]	5-Fold[1]	63.71 %
4	DML [26]	5-Fold[2]	71.03 %
5	Fusion CNN [27]	5-Fold	64.22 %
6	KinNet [17]	RFIW'17	74.85 %
7	AdvNet [18]	RFIW'17	66.58 %
8	LPQ-SIEDA [28]	RFIW'17	54.81 %
9	Multi-Set Learning [29]	RFIW'17	63.10 %
10	Parallel-SPCNN [30]	RFIW'17	61.33 %
11	FSP Classifier [10]	n/a	58.60 %
12	SelfKin [31]	RFIW'18	68.20 %
13	FA-CNN (Our)	5-Fold	67.94 %
14	Ensemble of FA-CNN (Our)	RFIW'17[3]	72.39 %
15	Ensemble of FA-CNN (Our)	RFIW'18	66.96 %
16	Ensemble of FA-CNN (Our)	5-Fold	68.84 %

[1] The tasks are face retrieval instead of verification

[2] Less face images comparing with standard 5-Fold training/testing split configuration

[3] Tested on validation set instead of testing set

RTX GPU with 24 GB RAM. PC with Intel i5-8400 @2.80 GHz equipped with 32 GB RAM was used to analyze the execution time on CPU. The execution time is taking by averaging 100 forward-pass of our proposed classifier. Table 5 shows the number of parameters of our proposed classifier along with the execution time on GPU and GPU. For GPU, we do not count the time for transferring the input image and the output of the classifier to from GPU.

As shown in Table 5, our proposed classifier can run around 9 fps using GPU and about 1 fps to 0.67 fps using CPU. Surprisingly, although the P-SphereFace variant classifier has more parameters, the execution time is lower than the P-FaceNet variant classifier which has a smaller number of parameters. Those phenomena appear because a huge number of parameters on the P-SphereFace variant classifier occurs in the fully-connected layers.

6 Comparison

For comparison, we cannot directly compare our classifier with other methods due to the different training/testing split configuration. Table 5 shows a comparison between our proposed classifier with several other methods. We also include the information of training/testing split configuration that used to evaluate each classifier to make fair information regarding the performance of the classifier. As shown in Table 5, our proposed classifier is comparable with other state-of-the-art methods and achieved 2nd best performance on all training/testing split configuration of the FIW dataset. Unfortunately, we cannot test the proposed model with RFIW 2017 test set because the submission server for the test evaluation is already closed by the organizer.

7 Conclusion

We have presented our proposed family-aware convolutional neural network (FA-CNN) for the visual kinship verification problem. Our proposed FA-CNN classifier constructed by taking state-of-the-art face recognition CNN architecture, freeze the weights and attaching layer for family-aware features learning with three additional fully-connected layers for the final classification decision. In short, we try to utilize deep metric learning features and combined with a softmax classifier to provide a probability output of the kin/non-kin category. Experiments on the FIW dataset show that our proposed classifier can achieve an average accuracy of 68.84% on the 5-fold configuration, 72.39% on RFIW 2017 dataset, and 66.96% on RFIW 2018 dataset.

An additional gating mechanism for the classifier may improve the performance of the classifier. One of the interesting questions is which area of the face is important or non-important regarding the visual kinship verification problems. Those phenomena can be used for gating the classifier and analyze the performance of the classifier based on the selected area of the faces.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

The main investigation, research, and analysis were provided by Reza Fuad Rachmadi and Supeno Mardi Susiki Nugroho. The funding acquisition and supervision were provided by I Ketut Eddy Purnama and Yoyon Kusnendar Suprapto. The manuscript is prepared by Reza Fuad Rachmadi and reviewed by all authors.

Acknowledgments

This research is partially supported by Ministry of Research and Higher Education of Indonesia under PDUPT scheme no. 930/PKS/ITS/2019

References

[1] J. Lu, J. Hu, X. Zhou, Y. Shang, Y.-P. Tan, and G. Wang, "Neighborhood repulsed metric learning for kinship verification", In: *Proc. of IEEE Conference On Computer Vision and Pattern Recognition (CVPR)*, Providence, USA, pp. 2594–2601, 2012.

[2] J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, and J. Zhou, "Neighborhood repulsed metric learning for kinship verification", *IEEE Transactions On Pattern Analysis and Machine Intelligence*, Vol. 36, No. 2, pp. 331–345, 2013.

[3] H. Yan and J. Hu, "Video-based kinship verification using distance metric learning", *Pattern Recognition*, Vol. 75, pp. 15–24, 2018.

[4] R. Fang, K. D. Tang, N. Snavely, and T. Chen, "Towards computational models of kinship verification", In: *Proc. of IEEE International Conf. On Image Processing* (ICIP), Hongkong, China, pp. 1577–1580, 2010.

[5] X. Qin, X. Tan, and S. Chen, "Tri-subject kinship verification: Understanding the core of a family", *IEEE Transactions on Multimedia*, Vol. 17, No. 10, pp. 1855–1867, 2015.

[6] J. P. Robinson, M. Shao, Y. Wu, H. Liu, T. Gillis, and Y. Fu, "Visual kinship recognition of families in the wild", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 40, No. 11, pp. 2624–2637, 2018.

[7] J. P. Robinson, M. Shao, Y. Wu, and Y. Fu, "Families in the wild (fiw): Large-scale kinship image database and benchmarks", In: *Proc. of ACM International Conf. on Multimedia*, New York, USA, pp. 242–246, 2016.

[8] S. Wang, J. P. Robinson, and Y. Fu, "Kinship verification on families in the wild with marginalized denoising metric learning", In: *Proc. of the IEEE International Conf. on Automatic Face and Gesture Recognition (FG)*, Washington, USA, pp. 216-221, 2017.

[9] M. B. López, E. Boutellaa, and A. Hadid, "Comments on the "kinship face in the wild" data sets", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 38, no. 11, pp. 2342–2344, 2016.

[10] M. Dawson, A. Zisserman, and C. Nellåker, "From same photo: Cheating on visual kinship challenges", In: *Proc. of Asian Conf. on Computer Vision*, Perth Western, Australia, pp. 654–668, 2018.

[11] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering", In: *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Boston, USA, pp. 815–823, 2015.

[12] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition", In: *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Hawai, USA, pp. 212–220, 2017.

[13] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition", In: *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, California, USA, pp. 4690–4699, 2019.

[14] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification", *IEEE Signal Processing Letters*, Vol. 25, No. 7, pp. 926–930, 2018.

[15] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille, "Normface: l2 hypersphere embedding for face verification", In: *Proc. of ACM International Conf. on Multimedia*, New York, USA, pp. 1041–1049, 2017.

[16] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition", In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Utah, USA, pp. 5265–5274, 2018.

[17] Y. Li, J. Zeng, J. Zhang, A. Dai, M. Kan, S. Shan, and X. Chen, "Kinnet: Fine-to-coarse deep metric learning for kinship verification", In: *Proc. of the 2017 Workshop on*

Recognizing Families In the Wild (RFIW), New York, USA, pp. 13–20, 2017.

[18] Q. Duan and L. Zhang, "Advnet: Adversarial contrastive residual net for 1 million kinship recognition", In: *Pro. of the 2017 Workshop on Recognizing Families In the Wild (RFIW)*, New York, USA, pp. 21–29, 2017.

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

[20] O. M. Parkhi, A. Vedaldi, A. Zisserman et al., "Deep face recognition", In *Proc. of British Machine Vision Conference (BMVC)*, Swansea, UK, pp. 41.1-41.12, 2015.

[21] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition", In: *Proc. of European Conference on Computer Vision (ECCV)*. Amsterdam, The Netherlands, pp. 499–515, 2016.

[22] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding", In: *Proc. of ACM International Conference on Multimedia*, Orlando, USA, pp. 675–678, 2014.

[23] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning", In: *Proc. of International Conference on Machine Learning*, Atlanta, USA, pp. 1139–1147, 2013.

[24] S. Wang, Z. Ding, and Y. Fu, "Cross-generation kinship verification with sparse discriminative metric", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 41, No. 11, 2018.

[25] Y. Wu, Z. Ding, H. Liu, J. Robinson, and Y. Fu, "Kinship classification through latent adaptive subspace", In: *Proc. of IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, Xi'an, China, pp. 143–149, 2018.

[26] S. Wang, J. P. Robinson, and Y. Fu, "Kinship verification on families in the wild with marginalized denoising metric learning", In: *Proc. of the IEEE International Conference on Automatic Face Gesture Recognition (FG)*, Washington, USA, pp. 216–221, 2017.

[27] R. F. Rachmadi, I. K. E. Purnama, S. M. S. Nugroho and Y. K. Suprapto, "Image-based Kinship Verification using Fusion Convolutional Neural Network", In: *Proc. of the IEEE International Workshop On Computational Intelligence and Applications (IWCIA)*, Hiroshima, Japan, pp. 59-65, 2019.

[28] O. Laiadi, A. Ouamane, A. Benakcha, and A. Taleb-Ahmed, "Rfiw-17: Lpq-sieda for large scale kinship verification", In: *Proc. of the 2017 Workshop on Recognizing Families in the Wild (RFIW)*, New York, USA, pp. 37–39, 2017.

[29] E. Dahan, Y. Keller, and S. Mahpod, "Kin-verification model on fiw dataset using multi-set learning and local features", In: *Proc. of the 2017 Workshop on Recognizing Families in the Wild (RFIW)*, New York, USA, pp. 31–35, 2017.

[30] R. F. Rachmadi and I. K. E. Purnama, "Paralel spatial pyramid convolutional neural network untuk verifikasi kekerabatan berbasis citra wajah", *Jurnal Teknologi Dan Sistem Komputer*, Vol. 6, No. 4, pp. 152–157, 2018.

[31] E. Dahan and Y. Keller, "Selfkin: Self adjusted deep model for kinship verification", *arXiv preprint* arXiv:1809.08493, 2018.