Tri-subject Kinship Verification using Feed Forward Neural Networks

G050 (s1674417, s1627278, s1654111)

Abstract

The development of computer vision coupled with an increase in facial image data has lead researchers to question whether it is possible to determine kinship, based merely on this data. We hypothesize that feed forward neural networks are a valid approach to solving tri-subject kinship verification. This paper uses the popular Families in the Wild (FIW) dataset in a tri-subject verification task which identifies mother, father and child relationships. The aim of this study is to compare four, state of the art, facial recognition networks: VGGFace, FaceNet, SphereFace and ArcFace. These models produce feature vectors as an output which are then channeled into a feed forward classifier. In comparing the feed forward networks created from these facial recognition models, we conclude that the feed forward network using features produced by FaceNet produces the highest area under curve (AUC) score of 0.731. Ensemble methods were incorporated into our analysis as our research found this approach to be beneficial in kinship verification tasks. The relevant ensemble classifiers methods are: Majority vote, Cascade Network, Highest Score and Threshold Majority Vote. The best ensemble method was Threshold Majority Vote which increased the AUC score to 0.756.

1. Introduction

Biologists find that human facial appearance is an important cue for genetic similarity measurement, therefore, visual kinship recognition allows us to predict these relationships (Dal Martello & Maloney, 2010; DeBruine et al., 2009). Visual kinship verification is the task of using facial images to determine the kinship status of a group of individuals. Within the area of visual kinship recognition there are several sub-tasks that have been proposed to enhance the variety of applications. The particular sub-task this paper will address is Tri-subject verification. Tri-subject verification takes the facial images of a set of biological parents and determines whether a given child is a blood relative based on their facial features. Visual Kinship verification has a wide range of practical applications that span multiple industries including forensic investigation, missing children, issues of human trafficking, historic lineage, social-media analysis, and immigration (Guo & Wang, 2012; Xia et al., 2012; Qin et al., 2015) to name a few.

Kinship verification was originally attempted using bi-subject kinship verification, i.e Mother-Child and Father-Child, which showed promise however could be greatly improved by the addition of the second parent (Qin et al., 2015). Recently, Northeastern University's Smile Labs proposed the 2020 Recognizing Families in the Wild (RFIW2020) challenge (Robinson et al., 2020) which seeks to determine the state-of-the-art in three key areas of visual kinship verification. One of these areas was that of tri-subject verification. The winning team, which is now regarded as state-of-the-art, was Team Ustc Nelslip. They proposed a Siamese network with ResNet50 and SENet50 as the backbone of the model, both these networks have been pre-trained on the VGGFace2 dataset (Cao et al., 2018). The second area was bi-subject kinship verification. This task used 13 different relationship types in their dataset, not only parent child relationships. Team Ustc Nelslip's tri-subject classifier achieved an accuracy score of 0.79 while the best bi-subject verification model achieved 0.78 on parent-child classification.

The focus of our research is to determine the validity of training feed forward neural networks using the feature vectors produced by four state-of-the-art facial recognition models for tri-subject kinship verification.

We hypothesised that the facial recognition models will provide enough discriminatory information to perform tri-subject verification using feed forward neural networks. We also predict that the use of different facial recognition models will lead to various facial features being captured resulting in differing performance.

This paper contributes to the limited existing research in the area of tri-subject verification. We discuss how the use of different loss functions can change the information encoded in feature vectors. This allowed us to create feed forward networks which were successful at correctly identifying relationships. Furthermore, we found that this information can be complimentary when applied to an ensemble model.

This paper is organised with **Section 2** which describes the dataset and the kinship recognition task. Next, **Section 3** is an overview of the methodologies used in this study. Then, **Section 4** coves our experiments and results. Afterwards, our discussion of results and future works are outlined in **Section 5**. Finally, our conclusions is in **Section 6**

2. Data-set and Task Evaluation

2.1. Families in the Wild dataset

The goal for FIW was to collect around 10 photos for 1,000 families, each with at least 3 family members. Below we present a brief description of the dataset:

- Number of unique faces: 30,000
- Number of families: 1000
- Average number of members per family: 4.927
- Average number of pictures per individual: 5.344
- Image size: 244 x 244 x 3

Further information regarding the data can be found at https://web.northeastern.edu/smilelab/fiw/

2.2. Related Work

(Robinson et al., 2018) looks at both Family Classification and Kinship Recognition as well as extending the Faces in the Wild dataset. They found that Res-Net 22 + Centerface is better than Sphereface at Family Classification while Sphereface performed best for Kinship Recognition. (Liu et al., 2017) looks at using Angular margin loss combined with Softmax Loss, this method outperformed the usual Euclidean loss approach for facial recognition. They explain how A-Softmax loss makes the decision regions become more separated, by enlarging the inter-class margin and compressing the intra-class angular distribution.

(Qin et al., 2015) was the very first attempt in tri-subjects kinship verification. The paper proposed a new discriminative bilinear classifier that models the similarity between the parents and child, with the dependence between them captured by a covariance matrix. The authors took forward the notion that images from both parents could provide richer information about the kinship relation to a child, since genetic overlapping between both parents and child would exist. The paper devised a vote-based feature selection method, which jointly selected the most discriminative features for the parents-child pair, while taking local spatial information into account - this method achieved state of the art results in this field.

The work done in (Robinson et al., 2020) is the main inspiration for our project. Recognizing Families In the Wild (RFIW) challenge series, is a large-scale data challenge consisting of multiple tasks with the aim to advance kinship recognition technologies. The aim of the RFIW challenge is to bridge the gap between research-and-reality using its large scale, variation, and rich label information. This paper created the framework for a global network of people to attempt to solve the tri-subjects kinship verification problem and showcase their submissions. The paper outlines the recommended data-splits, settings and metrics as well. In accordance with the RFIW challenge, we have chosen to follow this as recommended. Finally, the paper outlines a baseline result in which a score is assigned to each triplet (F_i, M_i, C_i) in the validation and test sets using the formula shown below:

$$\text{score}_{\,i} = \text{avg}\left(\cos\left(F_i, C_i\right), \cos\left(M_i, C_i\right)\right) \tag{1}$$

where F_i, M_i and C_i are the feature vectors of the father, mother, and child images respectively from the i-th triplet. Scores are then compared to a threshold γ to infer a label i.e., kin or non-kin.

2.3. Task Evaluation using AUC

The receiver operating characteristic curve (ROC), is a plot of the true positive rate (TPR) versus the false positive rate (FPR) for the predictions of the binary classifier at multiple thresholds. The integrated area under the curve (AUC) provides a summary measure of the discriminative ability of the model across all evaluated thresholds. AUC is desirable for the following two reasons:

- AUC is scale-invariant. It measures how well predictions are ranked, rather than their absolute values.
- AUC is classification-threshold-invariant. It measures the quality of the model's predictions irrespective of what classification threshold is chosen.

The AUC of the ROC curve, corresponds to the value of the WilcoxonMann-Whitney test, it is used as "a measure of goodness for predictions" (Vihinen, 2012). The range of AUC ROC values is between 0.5 and 1.0 with a value of 0.5 representing a classifier that is no better that randomly guessing the class and a value of 1.0 signifying a classifier with perfect discriminative ability. Hence, AUC provides a better performance metric as compared to the accuracy of the model and has been used as the default metric to assess the different models used in this study.

3. Methodology

The start of the section reviews the state-of-the-art face recognition models that we have used, by discussing their loss functions and the reason behind their success. Next, we talk about what an ensemble method is and what approaches currently exist. Finally, we will discuss our data separation, data pre-processing and our baseline experiment.

3.1. VGGFace

This model is a Convolutional Neural Network using ReLU activation layers and max pooling. VGG-Face contains 13 convolutional layers followed by 3 fully connected linear layers. We have chosen to adapt this CNN for our task as it has widely been used in benchmarking and novel image detection studies (Huang et al., 2008; Parkhi et al., 2015). These studies report that the architecture is able to deal well with facial image data and provides robust facial embeddings.

In VGG-Face, each convolutional layer has a kernel of size 3x3 with a padding and stride of 1. The CNN can be broken into blocks with layers of size of 2, 2, 3, 3 and 3. A more detailed overview of the VGG-Face hierarchy can be found

Layer	Input	Output
Layer 1	3	64
Layer 2-3	64	64
Layer 4-5	128	128
Layer 6-7	256	256
Layer 8	256	512
Layer 9-13	512	512
Layer 14	512*7*7	4096
Layer 15	4096	4096
Layer 16	4096	4096

Table 1. VGG-Face layer hierarchy. With Block and Layer numbers as well as their number of input and output channels

in Table 1. Max pooling with a kernel size of 2 x 2 and a stride of 2 is performed between each block of layers in order to reduce the number of parameters that require training in the network.

In between the thirteenth and fourteenth layer the output produced by the final convolutional layer is flattened in order to be used as an input to the fully connected linear layers. The network also applies dropout, to improve generalization, in between the fully connected layers by randomly assigning the value zero to certain elements of the input tensor with probability of 0.5 during training (Hinton et al., 2012).

The VGG-Face model was trained on the VGG-Face dataset (Parkhi et al., 2015) which was created and released by Oxford University. The output of the final fully connected layer, also known as the classification layer, is a vector of size 2,622. The final layer is used to classify images as one of 2,622 participants used to train the network. (Parkhi et al., 2015) suggests that by removing the last layer, the output of the layer below can be used as a feature vector. Furthermore, this feature vector can be used for face identity verification by calculating the Euclidean distance between images.

3.2. FaceNet

Our first facial recognition model we will analyse is FaceNet (Schroff et al., 2015), proposed by Google. FaceNet consists of two different core architectures - The Zeiler and Fergus (Zeiler & Fergus, 2014) network and the 2015 Inception (Szegedy et al., 2015) network. The power of the model lies in its ability to perform end-to-end learning for the whole system. Figure 1 shows a black box view of FaceNet's structure. It showcases how the network is trained to directly optimize the embedding itself, rather than an intermediate bottleneck layer as in previous deep learning approaches. The loss function used in FaceNet is triplet loss. As shown in Figure 2 the objective of this loss function is to increase intraclass distance while decreasing interclass distance. This means that the squared distance between pictures of an individual should smaller than the distance between pictures of others, making it easier to correctly classify input images.

CNN	Decision Boundary
Softmax Loss	$(W_1 - W_2)\,\boldsymbol{x} + b_1 - b_2 = 0$
SphereFace	$\|x\|\,(\cos m\theta_1 - \cos\theta_2) = 0$
ArcFace	$s\,(\cos(\theta_1 + m) - \cos\theta_2) = 0$

Table 2. Comparison of decision boundaries in binary case.

According to (Schroff et al., 2015), triplet loss's goal is to enforce a pairwise margin between faces from one person to all other individuals. This allows the faces for one identity to live on a manifold, while still enforcing intraclass distancing and thus increasing the networks discriminative powers. This model seems to be promising after it achieved an accuracy of 99.63% in a face recognition task (Sun et al., 2015).

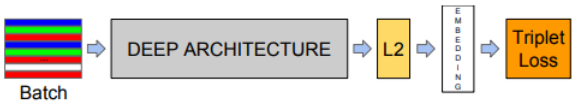

Figure 1. Model structure. The network consists of a batch input layer and a deep CNN followed by L2 normalization, which results in the face embedding. This is followed by the triplet loss during training.

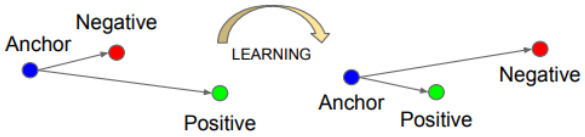

Figure 2. The Triplet Loss minimizes the distance between an anchor and a positive, both of which have the same identity, and maximizes the distance between the anchor and a negative of a different identity.

3.3. SphereFace

The second facial recognition model is SphereFace which was introduced in 2017. This CNN was unique as it proposed using an angular softmax loss (A-Softmax) function. The result of using this loss function lead to angularly discriminative feature vectors as shown in Figure 3. And through optimizing this loss function, the decision regions become more separated by simultaneously enlarging the inter-class margin while reducing the intra-class angular distribution (Liu et al., 2017).

A-Softmax loss defines a large angular margin (m) learning task with adjustable difficulty. With a larger m, the angular margin increases, the constrained region on the manifold becomes smaller, and the corresponding learning task also becomes more difficult.

3.4. ArcFace

The third facial recognition model is ArcFace which was introduced in 2018(Deng et al., 2019). ArcFace implements an additive angular margin loss function. This is similar to

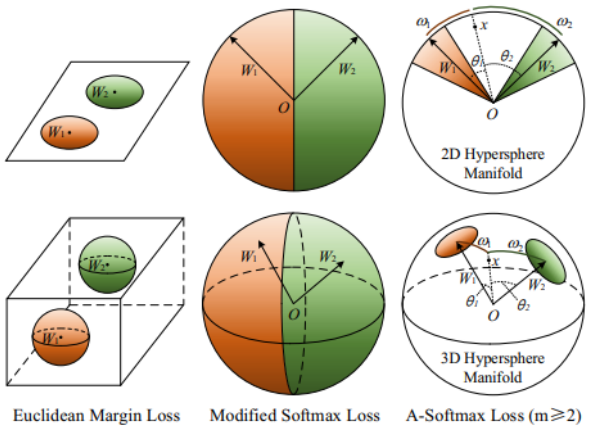

Figure 3. Geometry Interpretation of Euclidean margin loss (e.g. triplet loss), modified softmax loss and A-Softmax loss. The row above represents a 2D feature constraint, and the row below is a 3D feature constraint. The orange region indicates the discriminative constraint for class 1, while the green region is for class 2.

SphereFace, which was the first facial recognition model to implement an angular loss margin function but required many approximations. These approximations often result in unstable training. ArcFace attempts to address this problem. By using an adaptive A-Softmax function ArcFace avoids the approximations that caused SphereFace's unstable training.

ArcFace has the following advantages (Deng et al., 2019):

- ArcFace optimises the geodesic distance margin by virtue of the exact correspondence between the angle and arc in the normalised hypersphere.
- ArcFace is highly effective as it achieves state-of-the-art performance on ten face recognition benchmarks including large-scale image datasets.
- ArcFace does not need to be combined with other loss functions in order to have stable performance, and converges on most training datasets.
- ArcFace only adds trivial computational complexity during training.

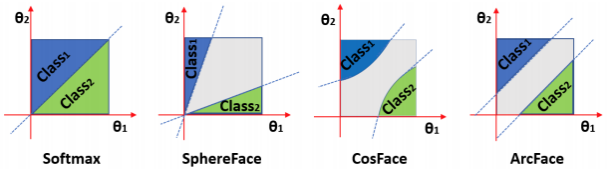

Figure 4. Decision margins of different loss functions under binary classification case. The dashed line represents the decision boundary, and the grey areas are the decision margins.

3.5. Success Seen with State of the Art Networks

FaceNet (Schroff et al., 2015) claims that, in 2015, it represents the most-accurate approach to recognizing human faces. In the popular MegaFace challenge FaceNet achieved an accuracy of 70.49% on the Face Identification task and an accuracy of 86.47 % on a Face Verification task. The MegaFace dataset (Miller et al., 2015) was recently released as a way to evaluate the performance of face recognition models. The dataset itself contains more than 1 million images from 690K different individuals. MegaFace ranks algorithms on two tasks - Face Verification and Face Identification.

The success seen by Google's algorithm is attributed to the company's private training data that contains several million identities. Face recognition models from industry perform much better than models from academia this is due to their significantly larger training sets. The enormity of certain training sets also makes some deep face recognition model's results not fully reproducible.

SphereFace was the first to apply A-Softmax loss to a CNN in order to learn discriminative facial features, it proved to be very successful on the MegaFace Challenge and succeeded in outperforming FaceNet. On the Face Identification task, SphereFace achieved an accuracy of 72.73% , when using a single model and an accuracy of 75.77% , when using a 3 patch ensemble. For the Face Verification task, SphereFace achieved an accuracy of 85.56% , when using a single model and an accuracy of 89.14% , when using a 3 patch ensemble.

At the time of writing, Arcface is the #1 CNN model on facial data, achieving an accuracy of 98.48% on the Face Verification task and an accuracy of 98.35% on the Face Identification tasks.

We have therefore chosen these models due to their high performance not only in the MegaFace challenge but also on the Labeled Faces in the Wild dataset found in Table 3, as well as the fact that they use different approaches when discriminating between users. This means that the feature vectors returned from these models have a higher chance of containing complementary data which hypothetically should improve the performance of our ensemble models.

Task	Model	Accuracy	Global Rank
Face Verification	FaceNet	99.63%	3
Face Verification	SphereFace	99.42%	6
Face Verification	ArcFace	99.83%	1

Table 3. Performance comparison on Labeled Faces in the Wild. dataset. VGG-Face does not appear in the top 10 methods for this dataset and is therefore not included in this table.

3.6. Ensemble methods

A common approach to improving regression and classification tasks is through the use of ensemble methods. Ensemble methods take a set of learning machines and combine them in order to obtain more reliable and accurate predictions (Re & Valentini, 2012). This approach stems from the idea of seeking several opinions before making a decision. Not only does this make sense intuitively but there has been a large degree of research (Ding et al., 2010; Tsai et al.,

2011; Jovanović et al., 2015) into why ensemble methods increase reliability and accuracy. With the research concluding that predictions provided by ensemble methods are often more accurate than those provided by a single classifier(Ren et al., 2016).

The initial reason to research the use of ensemble methods was based on the fact that all teams in the competition combined at least two CNN's to perform classification. Upon further research it showed that statistically ensemble methods can improve both regression and classification tasks. Training algorithms can often get stuck in a local minima. Through the use of ensemble methods we achieve the following results, in the best case scenario merging a set of suboptimal classifiers may achieve a better approximation leading to an improved accuracy whereas in the worst case scenario it will at least avoid the worst local minima performance (Re & Valentini, 2012).

We are considering using non-generative ensemble methods, these are models were the learning machines have already been selected and trained. There are two typical approaches for non-generative ensemble models: ensemble fusion and ensemble selection. Ensemble fusion methods use the output from all the base classifiers in order to arrive at a classification. Ensemble selection on the other hand identifies the best classifiers from a set of base classifiers. This forms a model specific to a given input which will be used to determine its classification.

The first ensemble fusion method we will consider is a Highest Score classifier (Caruana et al., 2004). We have decide to implement this model as it is similar to the unweighted average classifier, which is one of most widely used approaches (Ju et al., 2017). This is due to its ease of implementation and the improvements it offers for a diversity of applications. The highest score classifier works by summing the probabilities of each class produced by the base classifier. Predictions are made based on the class with the largest summed probability.

The second ensemble fusion model that we implemented was a Majority Vote Ensemble method (Matan, 1996; Brown & Kuncheva, 2010). This approach takes the mode of all the base classifiers predictions and uses this as the ensembles classification. Majority voting was the approach adopted by the winning team of the tri-subject verification challenge in the 2020 RFIW competition (Robinson et al., 2020), the winning team call it the jury system which is based of the Condorcet's jury theorem (Austen-Smith & Banks, 1996). This theorem takes the assumption that a set of competent jurors, in our case classifiers, each with an accuracy of over 50% will improve prediction accuracy by using majority voting.

The first ensemble selection model we opted to implement was the Cascading Classifier Ensemble (Zhang et al., 2007). This approach works by sequentially applying each base classifier starting with the classifier that performed best on the validation set during training. If the current classifier's confidence is above a given threshold then this classifier's predicted class is the output of the ensemble model. This process is repeated with the next best base classifier and continues until the ensemble model arrives at a classification.

The second ensemble selection method is called Threshold Majority Vote. Inspired by the ensembles seen in (Sidhu & Bhatia, 2018), this method takes all the base classifiers with a confidence score above a given threshold and uses this subset to perform majority voting. If this is inconclusive then majority voting is applied to the entire group of classifiers. However due to the fact that we use four base classifiers this may also lead to inconclusive results. In this situation we predict the class outputted from the base classifier that performs best on the validation set during training.

3.7. Data

3.7.1. Data Separation

To conduct our experiments, we decided to split our original data set into a training, validation and a test set. The splits were done as follows:

- Training Set: 55%
- Validation Set: 22.5%
- Test Set: 22.5%

Additionally, we ensured that there was no family-overlap between the three sets. Robinson et al. (Robinson et al., 2016) details the need to ensure that the dataset splits are balanced. This is done as follows - for each triplet (mother, father, and child), we add a non-kin triplet by keeping the mother and father the same while using a random child from a different family. A 50/50 split between kin and non-kin would likely result in better model performance for a binary classification task (Wei & Dunbrack Jr, 2013).

3.7.2. Data Pre-processing

Initially, a set of all images in the dataset was composed. Next, each image was cropped and normalised as done in (Robinson et al., 2016; Kazemi & Sullivan, 2014), and then resized to the networks input requirements. The image is then pipe-lined into the chosen network to produce a feature embeddings. We then applied l2 normalisation to the feature embeddings in order to reduce the likelihood of being stuck in local optima and also reduce the time training with the chance of better results (Sola & Sevilla, 1997).

3.8. Baseline

(Robinson et al., 2020) created a baseline using SphereFace and we have followed suit. The paper outlines a baseline result which assigns a score to each triplet (F_i, M_i, C_i) in the validation and test sets using the formula shown below:

$$\text{score}_{\,i} = \text{avg}\left(\cos\left(F_i, C_i\right), \cos\left(M_i, C_i\right)\right) \tag{2}$$

where (F_i, M_i, C_i) are feature vectors for the father, mother

and child respectively from the i-th triplet.

Scores were then calculated using the validation set and a threshold γ was calculated to get the best AUC score. We used $\gamma = 0.8875$. Using γ we then calculated predictions for the test set. If score $< \gamma$ the triplet was classified as a relation otherwise a non-relation. The results of the baseline can be seen in Table 5

Although (Robinson et al., 2020) reported a baseline score of 0.68 and we have a score of 0.612, our method correctly splits the dataset. We have implemented their procedure accordingly, but could not compare the splits due to an error in the current data release.

4. Experiments

4.1. Experimental Settings

We opted to use Feed Forward Neural Networks in order to perform classification using the feature vectors produced by the different Facial Recognition models. This decision was due to the fact that feed forward layers work well on simple input data as seen in (Rychetsky et al., 1998; Wang, 2002). We created a feed forward neural network for each of the following facial recognition models: ArcFace, FaceNet, SphereFace and VGGFace. These feed forward networks have an input of size 1536, as each facial recognition model produces a feature vector of size 512×1 for every face it takes as input. Since we are using three facial images; mother, father and potential child; we get a 1 dimensional tensor of size 3×512 as input. We have an output of size 2, one node for related and another for not related.

Since there is no default approach on constructing a feed forward neural network, we decided to take influences from past literature. The winning team for the tri-subject verification task in the 2020 RFIW challenge was Team Ustc-Nelslip (Robinson et al., 2020). Their approach involved an ensemble classifier with two different facial recognition models, the backbone of these models are: SENet50 and ResNet50. They also employed two loss functions, binary cross entropy and focal loss. Therefore, we began our exploration using a binary cross entropy loss function.

We began by running experiments on each of the face recognition models, to find the highest AUC score on our validation set. Our initial approach was to find the optimum number of layers in our feed forward network. It quickly became apparent that deeper networks tend to overfit to the training data resulting in a very high training accuracy and small training loss. As a result the network found it difficult to generalise to unseen data and performed poorly on the validation set. Shallower networks on the other hand, resulted in higher AUC scores. Shallow networks have smaller amount of parameters and due to this reduced complexity, shallow networks have a higher generalizability.

Next, we experimented with a varying number of nodes per hidden layer and found that the structures seen in Table 4 performed best using their respective facial recognition model's feature vectors.

Facial Recognition	Hidden nodes	Activation Function
ArcFace	1024 -> 512	ReLU
FaceNet	1024 -> 512	ReLU
SphereFace	2048	PReLU
VGGFace	1024 -> 64	ReLU

Table 4. Feed Forward Neural Networks.

Network	FMD	FMS	AUC score
Sphereface (baseline)	0.614	0.610	0.612
ArcFace	0.719	0.721	0.720
Sphereface	0.634	0.637	0.636
VGGFace	0.711	0.730	0.721
FaceNet	**0.765**	**0.761**	**0.763**

Table 5. Performance of feed forward networks and baseline. FMD is the father, mother and daughter relationship. FMS is the father, mother and son relationship. All results are AUC scores.

Fixing network depth and the number of nodes per hidden layer, we then began to alter the loss functions used when training each network. We expected binary cross entropy loss to perform best as this is the loss function that Team Ustc-Nelslip used to win the 2020 RFIW tri-subject kinship verification challenge. The results from our experiments were aligned with our initial assumption and hence we use Binary Cross Entropy Loss to train all our networks.

The final aspect of the architecture left to explore was the activation functions applied to each layer. We decided to reduce the scope of experimentation due to time constraints by ensuring that all layers applied the same activation function. As shown in Table 4 the majority of feed forward neural networks performed optimally using ReLU activation function whereas SphereFace performed better with PReLU with $\alpha = 0.25$.

The results in Table 5 highlight the best AUC scores on the validation set which were achieved during training using the facial recognition networks described in Table 4. As shown by the table the network with the highest AUC score was FaceNet. We then applied FaceNet to our test set which gave us an **AUC score of 0.731**.

4.2. Ensemble Experiments

The ensemble experiments were done to investigate whether using an ensemble method would further increase our AUC score. This is important because we hypothesised that different facial recognition networks capture different features of a persons face. An ensemble would allow us to potentially reap the benefits of the face recognition models together. The results of our experiments can be seen in Table Table 6.

After experimenting with different ensemble methods and their thresholds values. The Threshold majority vote achieved the best score on the validation set. We then proceeded to apply the Threshold Majority Vote to our test

Ensemble Method	Threshold	AUC Score
Majority vote	N/A	0.753
Cascade network	0.98	0.761
Highest score	N/A	0.762
Threshold Majority vote	**0.85**	**0.77**

Table 6. Performance of ensemble methods and their threshold values. The results are using the validation set.

set which gave us an **AUC score of 0.756**.

5. Discussion

This section of our report aims to evaluate and critique the results we obtained using feed forward networks and ensemble methods in the context of tri-subject verification. We begin by discussing our results from the feed forward neural networks and propose future works in this field, followed by the same discussion for our Ensemble Methods.

5.1. Feed Forward Neural Network

Our experimentation in the feed forward layer included comparing different activation functions and different loss functions. However, these changes have the potential to affect the optimal hidden layers and nodes. In our case, we chose the best configuration given the time constraint on the number of experiments that we could reasonably complete. We feel that rigorous experiments on hyperparamater tuning could lead to further improvements. We can asume that these improvements would be minor in nature as altering the architecture, the loss function and activation function during testing lead to minimal change in validation accuracy.

Having examined past literature on kinship classification and verification, we suggest that experiments with Maxout neuron (introduced recently by Goodfellow (Goodfellow et al., 2013)) as an activation function could further generalizes the ReLU and its leaky version. The Maxout neuron computes the function $\max\left(w_1^T x + b_1, w_2^T x + b_2\right)$. The ReLU and Leaky ReLU are a special case of this form (ReLU has $w_1, b_1 = 0$). The Maxout neuron activation therefore enjoys all the benefits of a linear regime of operation and no saturation of a ReLU unit. Similarly, it does not have its drawbacks, in particular, dying ReLU. Maxout has been used in previous face verification CNN networks as seen in (He et al., 2017), where using this activation has allowed the network to perform better and overcome the problem of the large number of "dead" units in the network. The potential drawback with this function is that unlike ReLU, using Maxout doubles the number of parameters for every single neuron and this could lead to an significant increase in the total number of parameters.

From our research we noticed that the most significant improvements can be made through altering the loss function. The negative log-likelihood or NLL has been proven to help boost the performance and face recognition capabilities, as observed in (Dunne & Campbell, 1997; Liu et al., 2015). The negative log-likelihood function produces a high loss when the values of the output layer are evenly distributed and low. In other words, there's a high loss value associated to unclear classifications. NLL also produces relatively high values when the classification is wrong. When the value of the output layer matches that of the expected value, the negative log-likelihood function produces a very low value. This would help the model converge better and would be great option to consider in future experiments.

Table 5 shows that FaceNet results in the highest AUC score. One reason for FaceNet being able to outperform the other algorithms is the triplet loss function that it uses. In triplet loss, a baseline input is compared to a positive input and a negative input. The distance from the baseline input to the positive input is minimized, and the distance from the baseline input to the negative input is maximized. Triplet loss is good at being able to increase intraclass distance while simultaneously decreasing interclass distance. A parallel can be drawn to the case of tri-subject kinship verification as Facenet would be able to better distinguish members that belong to the same family unit while separating those subjects that do not form a family. This form of loss function relatively simpler when compared to the loss functions used by ArcFace and SphereFace. Meaning triplet loss would have a lower chance of overfitting and greater generalizability.

Sphereface and Arcface use loss functions that involve a Multiplicative Angular Margin and a weight normalization step, therefore resulting in a more complex loss function. Both these networks were primarily designed with the focus of performing facial verification (Deng et al., 2019), a task that largely involves being able to re-recognise the same face in various scenarios. We can think of this as tuning a loss function in order to accomplish this specific task. Since our research is focused more towards being able to distinguish if a set of three images of a Father-Mother-Child constitutes a family unit or not. In this case, the need to re-recognise the same face is not as vital as the classier being able to draw boundaries between between the groups of images that are a family and the groups of images that are not. Hence, we can say that using the highly complex loss function would lead to poorer genralizability, and eventually result in a poorer performance.

5.2. Ensemble Methods

We chose to design an ensemble classifier to try and improve the performance achieved by the single models (base learners) by combining them together. The reason for this is that each model captures different aspects of an individual's face and its kinship relation. By combining these models we can create a more robust kinship verification model that can effectively capture tri-subjects units. Interestingly, the highest score from individual classifier, as seen in Table 5 is 0.763 and the experiments from the ensemble methods, in 3 out 4 cases resulted in a decrease in the AUC score. The only method to that successfully increased our score

was **"Threshold Majority Vote"**. The rational for Threshold Majority Vote outperforming the alternative methods is as such - classifying individuals as related or not related, would require (by casting a vote) a certain amount of "confidence" / high probability. This probability would have to be greater than the threshold value to vote ensuring that "low confidence" models do not interfere with confident ones. The Majority vote Ensemble is the lowest performing model from the chosen ensemble methods. This is because each model has the equal voting power. Therefore, "low confidence" models have just as much ability to sway the decision of the ensemble as model with a higher degree of confidence. The Cascade Network Ensemble did not perform well as it takes the best model's confidence to classify relationships which is detrimental if the model correctly identifies a relationship but with low confidence. Since these correct classifications could be missed, we see a worse score as compared to just using FaceNet. Finally, the Highest Score Ensemble takes the four models, adds each model's relationship and non relationship score and takes Model which has the maximum value. This is, similar to why Majority Vote performed badly, problematic as it gives each model equal weighting.

In the models that incorporated a threshold i.e., Cascade Networks and "Threshold Majority Vote" , experiments showed us that a lower threshold value lead to an decrease in the accuracy score. In the case of "Cascade Network" this means it is easier for predictions to be made which encapsulates the possibility of wrong predictions being made. Therefore, lowering the score. On the other hand, lowering the threshold in "Threshold Majority Vote" allows, comparatively, not confident models to vote.

Equal weighting of confidence between models poses a significant problem when distinguishing which model performs most accurately. This problem can be seen in the only slight 0.007 increase in accuracy scores from single Networks to multiple Network Ensemble methods.

5.2.1. Future works

Within the context of Ensemble Methods, the weight given to respective Models is important to take into consideration. Specifically, a higher weight should be given to the models that have a higher AUC score when tested on the validation set. Future research into this field should investigate how to weight the models effectively. For example, training a linear classifier which weights model based on how well they perform.

As part of a broader experimentation aspect, we would like to add CosFace - another high performing face recognition and verification CNN architceture that performs well on MegaFace and the Labeled Faces in the Wild data-sets. This could also potentially be another model that could be used in the ensemble classifier to increase the final AUC score.

Additionally, we hope to leverage other types of prior knowledge by learning multiple complementary features to better represent the facial data as embeddings in feature vectors. This would help in extending our framework to handle a more general family structure.

6. Conclusions

In this work, we made the attempt to investigate the Tri-subject kinship verification problem. Contrary to the well researched Bi-Subject Kinship Verification that uses information from a single parent-child relation, we exploited information from both parents to identify kinship between them and their child. For this we used four well recognised face detection models: FaceNet, SphereFace, VGGFace and FaceNet. We compared the performance of these models with each other as well as with ensemble methods that incorporated these four models.

Our hypothesis stated that the use of feed-forward networks will be able to discriminate features to perform tri-subject verification. We have found that this setting creates a credible use of feed forward networks. Moreover, we further increased performance by incorporating ensemble methods. This follows on from our hypothesis that various facial recognition networks capture differing features. We decided to take this approach due to the previous success seen from the 2020 RFIW challenge. The best ensemble method we found was "Threshold Majority Vote" which increased our AUC score even further when compared to the individual base classifiers.

Despite the seemingly high accuracy scores, feed food forward neural networks are just one approach to tri-subject verification. We suspect as tri-subject verification becomes more popular and bigger datasets are made available, an accurate end-to-end model could be created to determine kinship. In future work we hope to incorporate extensive experiments using additional high performing face recognition networks. By the same token, future work should investigate the best way to learn complementary features to better represent facial data. Although ensemble methods were not an original part of our hypothesis, future work should aim to integrate the benefits of such a method when creating a kinship classifier.

References

Austen-Smith, David and Banks, Jeffrey S. Information aggregation, rationality, and the condorcet jury theorem. *American political science review*, 90(1):34–45, 1996.

Brown, Gavin and Kuncheva, Ludmila I. "good" and "bad" diversity in majority vote ensembles. In *International workshop on multiple classifier systems*, pp. 124–133. Springer, 2010.

Cao, Qiong, Shen, Li, Xie, Weidi, Parkhi, Omkar M, and Zisserman, Andrew. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pp. 67–74. IEEE, 2018.

Caruana, Rich, Niculescu-Mizil, Alexandru, Crew, Geoff,

and Ksikes, Alex. Ensemble selection from libraries of models. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 18, 2004.

Dal Martello, Maria F and Maloney, Laurence T. Lateralization of kin recognition signals in the human face. *Journal of vision*, 10(8):9–9, 2010.

DeBruine, Lisa M, Smith, Finlay G, Jones, Benedict C, Roberts, S Craig, Petrie, Marion, and Spector, Tim D. Kin recognition signals in adult faces. *Vision research*, 49(1):38–43, 2009.

Deng, Jiankang, Guo, Jia, Xue, Niannan, and Zafeiriou, Stefanos. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4690–4699, 2019.

Ding, Jiandong, Zhou, Shuigeng, and Guan, Jihong. Mirensvm: towards better prediction of microrna precursors using an ensemble svm classifier with multi-loop features. *BMC bioinformatics*, 11(S11):S11, 2010.

Dunne, Rob A and Campbell, Norm A. On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function. In *Proc. 8th Aust. Conf. on the Neural Networks, Melbourne*, volume 181, pp. 185. Citeseer, 1997.

Goodfellow, Ian J, Warde-Farley, David, Mirza, Mehdi, Courville, Aaron, and Bengio, Yoshua. Maxout networks. *arXiv preprint arXiv:1302.4389*, 2013.

Guo, Guodong and Wang, Xiaolong. Kinship measurement on salient facial features. *IEEE Transactions on Instrumentation and Measurement*, 61(8):2322–2325, 2012.

He, Ran, Wu, Xiang, Sun, Zhenan, and Tan, Tieniu. Learning invariant deep representation for nir-vis face recognition. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Hinton, Geoffrey E, Srivastava, Nitish, Krizhevsky, Alex, Sutskever, Ilya, and Salakhutdinov, Ruslan R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

Huang, Gary B, Mattar, Marwan, Berg, Tamara, and Learned-Miller, Eric. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. 2008.

Jovanović, Radiša Ž, Sretenović, Aleksandra A, and Živković, Branislav D. Ensemble of various neural networks for prediction of heating energy consumption. *Energy and Buildings*, 94:189–199, 2015.

Ju, Cheng, Bibaut, Aurélien, and Laan, Mark. The relative performance of ensemble methods with deep convolutional neural networks for image classification. *Journal of Applied Statistics*, 45, 04 2017. doi: 10.1080/02664763.2018.1441383.

Kazemi, Vahid and Sullivan, Josephine. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1867–1874, 2014.

Liu, Fayao, Shen, Chunhua, and Lin, Guosheng. Deep convolutional neural fields for depth estimation from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5162–5170, 2015.

Liu, Weiyang, Wen, Yandong, Yu, Zhiding, Li, Ming, Raj, Bhiksha, and Song, Le. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 212–220, 2017.

Matan, Ofer. On voting ensembles of classifiers. In *Proceedings of AAAI-96 workshop on integrating multiple learned models*, pp. 84–88. Citeseer, 1996.

Miller, Daniel, Brossard, Evan, Seitz, S, and Kemelmacher-Shlizerman, Ira. Megaface: A million faces for recognition at scale. *arXiv preprint arXiv:1505.02108*, 2015.

Parkhi, Omkar M., Vedaldi, Andrea, and Zisserman, Andrew. Deep face recognition. In *British Machine Vision Conference*, 2015.

Qin, Xiaoqian, Tan, Xiaoyang, and Chen, Songcan. Tri-subject kinship verification: Understanding the core of a family. *IEEE Transactions on Multimedia*, 17(10): 1855–1867, 2015.

Re, Matteo and Valentini, Giorgio. *Ensemble methods: A review*, pp. 563–594. 01 2012.

Ren, Ye, Zhang, Le, and Suganthan, Ponnuthurai N. Ensemble classification and regression-recent developments, applications and future directions. *IEEE Computational intelligence magazine*, 11(1):41–53, 2016.

Robinson, J. P., Shao, M., Wu, Y., Liu, H., Gillis, T., and Fu, Y. Visual kinship recognition of families in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(11):2624–2637, Nov 2018. ISSN 1939-3539. doi: 10.1109/TPAMI.2018.2826549.

Robinson, Joseph P., Shao, Ming, Wu, Yue, and Fu, Yun. Families in the wild (fiw): Large-scale kinship image database and benchmarks. In *Proceedings of the 2016 ACM on Multimedia Conference*, pp. 242–246. ACM, 2016.

Robinson, Joseph P, Yin, Yu, Khan, Zaid, Shao, Ming, Xia, Siyu, Stopa, Michael, Timoner, Samson, Turk, Matthew A, Chellappa, Rama, and Fu, Yun. Recognizing families in the wild (rfiw): The 4th edition. *arXiv preprint arXiv:2002.06303*, 2020.

Rychetsky, Matthias, Ortmann, Stefan, and Glesner, Manfred. Pruning and regularization techniques for feed forward nets applied on a real world data base. In *NC*, pp. 603–609, 1998.

Schroff, Florian, Kalenichenko, Dmitry, and Philbin, James. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015. URL http://arxiv.org/abs/1503.03832.

Sidhu, Parneeta and Bhatia, MPS. A novel online ensemble approach to handle concept drifting data streams: diversified dynamic weighted majority. *International Journal of Machine Learning and Cybernetics*, 9(1):37–61, 2018.

Sola, J and Sevilla, Joaquin. Importance of input data normalization for the application of neural networks to complex industrial problems. *IEEE Transactions on nuclear science*, 44(3):1464–1468, 1997.

Sun, Yi, Wang, Xiaogang, and Tang, Xiaoou. Deeply learned face representations are sparse, selective, and robust. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2892–2900, 2015.

Szegedy, Christian, Liu, Wei, Jia, Yangqing, Sermanet, Pierre, Reed, Scott, Anguelov, Dragomir, Erhan, Dumitru, Vanhoucke, Vincent, and Rabinovich, Andrew. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

Tsai, Chih-Fong, Lin, Yuah-Chiao, Yen, David C, and Chen, Yan-Min. Predicting stock returns by classifier ensembles. *Applied Soft Computing*, 11(2):2452–2459, 2011.

Vihinen, Mauno. How to evaluate performance of prediction methods? measures and their interpretation in variation effect analysis. In *BMC genomics*, volume 13, pp. S2. BioMed Central, 2012.

Wang, XiaoHu. Assessing performance measurement impact: A study of us local governments. *Public Performance & Management Review*, 26(1):26–43, 2002.

Wei, Qiong and Dunbrack Jr, Roland L. The role of balanced training and testing data sets for binary classifiers in bioinformatics. *PloS one*, 8(7), 2013.

Xia, Siyu, Shao, Ming, Luo, Jiebo, and Fu, Yun. Understanding kin relationships in a photo. *IEEE Transactions on Multimedia*, 14(4):1046–1056, 2012.

Zeiler, Matthew D and Fergus, Rob. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

Zhang, Ping, Bui, Tien D, and Suen, Ching Y. A novel cascade ensemble classifier system with a high recognition performance on handwritten digits. *Pattern Recognition*, 40(12):3415–3429, 2007.