Facial Kinship Verification with Large Age Variation Using Deep Linear Metric Learning

Diego Lelis[1] and Dibio L. Borges[2]
[1] Department of Mechanical Engineering, University of Brasilia, Brasilia, Brazil
[2] Department of Computer Science, University of Brasilia, Brasilia, Brazil
Email: diego.o.lelis@gmail.com, dibio@unb.br

***Abstract*—Facial appearance affects how humans interact with each other. It is also how relatives are visually identified to determine how social interactions proceed. Humans can identify kin relations based only on the face. Intrinsically, giving the ability to detect kin relations to computers can improve their usefulness in our daily lives. This research proposes a solution to the kinship verification problem with a novel non-context-aware approach. The approach is validated using a dataset with large age variation upon which is applied the proposed Deep Linear Metric Learning (DLML). The method leverages multiple deep learning architectures trained with massive facial datasets. The knowledge acquired on traditional facial recognition tasks is re-purposed to feed a linear metric learning model. The proposed method was able to achieve better performance than other context-aware methods on tests that are inherently more difficult than ones used on previous methods with the UB Kinface dataset. The results show that the method can use the knowledge of deep learning architectures trained to perform mainstream facial recognition tasks with massive datasets to solve kinship verification on the UB Kinface database with robustness towards large age differences present on the dataset. The method also offers enhanced applicability when compared to previous methods in real-world situations, because it removes the necessity of knowing/detecting and treating large age variations to perform kinship verification. Additional tests were also performed at the KinFaceW-I and KinFaceW-II datasets to further assess performance of the method.**

***Index Terms*—kinship verification, deep learning, feature re-purposing, metric learning, facial recognition, convolutional neural network**

I. Introduction

Different from the most common facial recognition approaches that mostly try to compare similarity, kinship verification is more complicated to solve because people with dissimilar appearances can be kin and people with similar appearances can be non-kin at all [1]-[5]. For instance, on Fig. 1, pairs ab and c-d are non-kin similar people, this proximity between facial characteristics provides a complex challenge for facial recognition models because it is necessary to identify what features can signal a kin relation to avoid false positives like the ones that it could easily occur between pairs a-b and c-d, for example.

Despite the difficulties to perform kinship verification, humans can identify kinship relations at a higher rate than chance, but it is not clear how [6]. On the case considered here, we also added the additional factor of large age variations with the UB Kinface dataset [1].

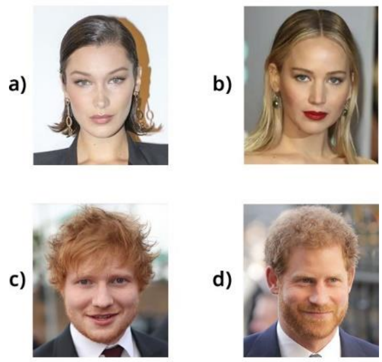


Figure 1. Images of similar non-kin people, a) to b), and c) to d) - source: [7]-[9].

Since the old parent's face structure is transformed when compared to when they were young [1], the age difference increases the distance between the face of child-old parent making it more difficult to identify the kin relation. On Fig. 2, it is possible to observe two examples of pairs of images (a-b and c-d) that because of the large age differences it would easily prompt a false negative if the model is based solely on the raw facial distance. The age difference present on the UB Kinface dataset makes the problem more challenging [1] and it has been treated separately by previous methods available on the literature [2].

A key factor that inspired this research is the fact that, to the best of our knowledge, all the other solutions for kinship verification with large age variations using the UB KinFace database, either try to preprocess the face of the old parent to approximate it to the child face as shown on Fig. 3, or trained the same method twice, one for the child-young parent pairs, and another for child-old parents pairs like on Fig. 4. On the case of the KinFaceW-I and KinFaceW-II [4], all other methods, to the best of our knowledge, are trained for each type of relation present on the dataset.

Our method is based on the approach exhibited in Fig. 5, and it offers a more practical and simple solution to all

Manuscript received January 8, 2019; revised May 13, 2019.

doi: 10.18178/joig.7.2.50-58

of these applications because it discards the need for detecting large age differences and different types of kin relations.

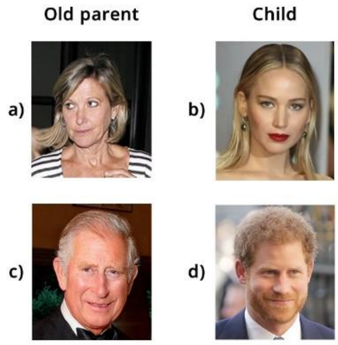


Figure 2. Images of kin people with large age differences - source: [8]-[11].

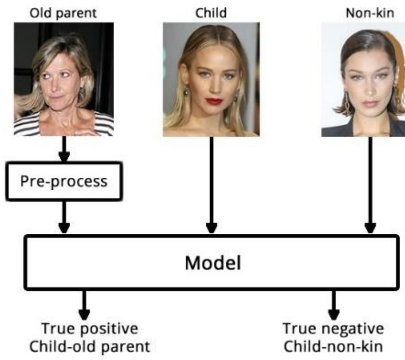


Figure 3. Pre-process strategy of previous methods that approximate the old parents face from the child face by reducing aging effects. Desired outputs are showed at last stage.

Our complete DLML proposed method is displayed on Fig. 6. The method is divided into four stages:

- **Face Alignment-MTCNN:** We detect and crop faces using a Multi-Task Convolutional Neural Network (MTCNN) [12]. The first phase will provide a picture of the face with 160x160 size as output.
- **Feature Extraction-FaceNet:** The processed images are then fed onto a FaceNet [13], [14] implementation that is going to generate embeddings of 128 dimensions of the face.
- **Feature subtraction:** The extracted features are subtracted to create an array of 128 dimensions that represents the distance between two faces.
- **Linear model:** Finally, the distance array of 128 dimensions is fed onto a linear model that is going to provide a boolean output informing if the two people are kin or not.

A. *Kinship Verification Applications*

- On passports checks because it is necessary to differentiate kin people. Kinship verification can be used to improve facial recognition models that are sensitive to this type of situation [2].
- Identifying the parents of lost children and orphans to help the work of law enforcement agencies [4].

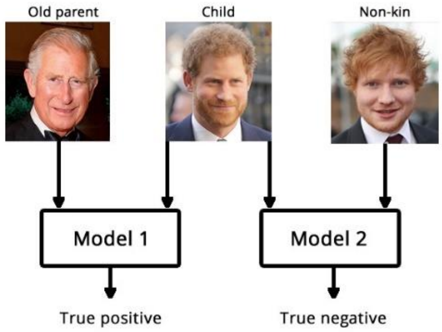


Figure 4. Different models strategy of previous methods that trained and tested the same method separately for child-young parent and child-old parent pairs with desired outputs.

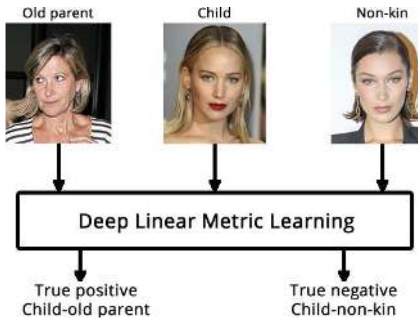


Figure 5. Approach of our method that it does not treat differently child-young parent and child-old parent pairs.

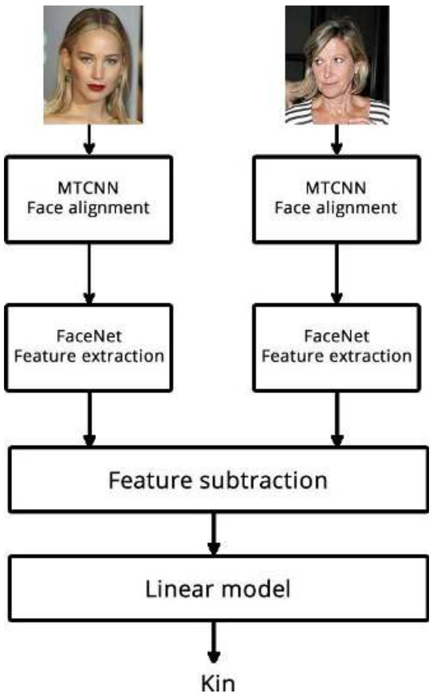


Figure 6. The complete proposed method to perform kinship verification.

- Improving target ads by using the preferences of their kin people to provide a more personalized experience [2].
- To organize family photos detecting kin relations on pictures.
- To search for relatives in public datasets [3].

- To allow make-up artists to modify the appearance of two people in a way that they seem blood-related [2].

B. The Contributions of This Research Are

- **Deep Linear Metric Learning:** Until now, all the past solutions have treated kinship verification with large age variations using the UB Kinface dataset as two separate problems, identify a child-young parent kin relation, and identify a kin relation between child-old parent. Our novel DLML method offers a new and more practical solution for kinship verification problem with large age variations, by using an all in one approach that enhances applicability on real-world situations. The proposed method also showed better than human performance in non-context-aware tests performed at KinFaceW-I and KinFaceW-II [4] datasets.
- **Feature re-purposing:** The results confirmed that the features extracted by our FaceNet model trained with VGGFace2 to perform facial recognition can be re-purposed to perform kinship verification with robustness towards large age variations present on the UB Kinface dataset by applying our linear metric learning approach. The extracted features also allowed to perform kinship verification at the KinFaceW-I and KinFaceW-II [4] datasets that have more gender diversity than the UB Kinface. This stage can also be called transfer learning.
- **Results:** The results provided by this research showed that the proposed DLML framework can identify kinship relations despite large age differences and with better performance than multiple other methods. Results at the KinFaceW-I and KinFaceW-II also showed that the proposed method is robust towards greater gender variation than the one present at the UB Kinface dataset and to low resolution images.

II. Related Work

Making kin annotations is more complicated than making annotations of identity because it is necessary to work with pairs. Inherently, it is more challenging to collect and annotate the data of datasets like UB Kinface, KinFaceWI, and KinFaceW-II, than the data of VGGFace2 that it is mainly used to detect identity. This complexity led to a scarcity in large kin-related datasets when compared to traditional datasets like Labeled Faces on the Wild (LFW) and VGGFace2 [2].

There is a consensus that the UB Kinface dataset is the kinship dataset with the largest age variations [2], however, the original paper [1] does not provide the values of the age differences between pairs. That is why the UB Kinface is the main dataset used on this research.

All the past solutions that used UB Kinface, KinFaceWI, and KinFaceW-II have focused mainly on achieving good results on these datasets, treating each type of pairs of images (child-young parent, child-old parent) or relations (father-daughter, father-son, mother-son, mother-daughter) like different problems [2]. The methods found in literature are difficult to apply in a real-world environment because they need to detect if there is a big age difference between the two faces or what is the gender of the subjects to decide what approach should be used.

To the best of our knowledge, Table I presents all the methods evaluated on the UB-kinface dataset. On Table I, the strategy column refers to how previous models were trained, the five-fold and leave-one-out columns on Table I are the average accuracy of these methods on child-young parents and child-old parents, unlike our method these evaluations are performed separately.

TABLE I. Results of Other Methods on the UB Kinface Dataset

Method	Five-fold	Leave-one-out	Strategy
fcDBN [3]	91.75%	Not tested	Different models
Visual Attr. [15]	82.50%	Not tested	Only child-old par.
DMML [5]	72.25%	Not tested	Different models
PDFL [16]	67.30%	Not tested	Different models
MNRML [4]	67.05%	Not tested	Different models
TL [17]	60.00%	Not tested	Pre-process
TSL [1]	56.50%	Not tested	Pre-process
SSRW [18]	53.90%	Not tested	Different models

Table II shows some of the most relevant results on KinFaceW-I and KinFaceW-II, all of these methods trained different models for each relation to maximize the accuracy, unlike our method that is trained to identify the relation despite gender and age.

TABLE II. Results of Other Methods with Five-Fold Cross-Validation on KinFaceW-I and KinFaceW-II

	Accuracy	
Method	KinFaceW-I	KinFaceW-II
Human [4]	63.75%	66.75%
fcDBN [3]	96.10%	96.20%
DMML [5]	72.25%	78.25%
PDFL [16]	70.05%	76.95%
MNRML [4]	69.90%	76.50%

Most attempts to solve kinship verification have used shallow machine learning methods like [19], [6], [5], [16], [17], [1], [18], [15], [4]. The only deep learning method evaluated on the UB Kinface dataset used about 600,000 images for train on feature extraction [3], more than five times less than our method that used more than three million images from VGGFace2.

To the best of our knowledge, the only deep learning method available on literature that performed kinship verification on the UB Kinface dataset is called filtered contractive Deep Belief Network (fcDBN) [3]. fcDBN is also the state of the art method for most of the publicly available datasets, like KinFaceW-I and KinFaceWII. fcDBN used for the first time external datasets to teach the model how to extract facial features to perform kinship verification [2]. fcDBN was tested on the UB Kinface dataset using five-fold cross-validation and achieved 92.00% of accuracy on child-young parent pairs, and 91.50% accuracy on child-old parent pairs [3], 91.75% on average as shown at Table I. fcDBN was also evaluated at KinFaceW-I and KinFaceW-II, the results are presented at Table II.

The research responsible for publishing the UB Kinface dataset [1] used a method called Transfer Subspace Learning (TSL) that uses local Gabor filters to extract features. These features are used to determine if parent and children have similar eyes, noses, or mouths. With the extracted key points, six ratios of common regions distances are obtained, e.g., eye-to-eye versus eye-nose distance. The TSL method performs context-aware tests reducing the divergence between child-old parent by using the child-young parent as an intermediate set. This research also published a human baseline of 56.00% of accuracy for kinship verification [1]. This human baseline shows that the problem tackled by this research is a difficult one.

Our method differs from fcDBN on the architectures used (MTCNN and FaceNet), on the dataset used to train the network how to extract facial features (VGGFace2). However, the main difference between our DLML and all the previous methods including fcDBN is the fact that our method can detect kin relations on the UB Kinface, KinFaceW-I and KinFaceW-II datasets without treating any age difference or gender variation, offering enhanced applicability.

Furthermore, in our case, to show how expressive the extracted features of our method are to perform kinship verification on the multiple datasets, our last stage is a simple linear artificial neural network.

III. Datasets

In this section, we explore the datasets used in the research and some of the necessary operations. All data used will be formed by unconstrained images (on the wild).

A. Training Datasets for Deep Learning

The VGGFace2 is a large-scale face dataset with large age and gender variations, composed of 3.31 million images of 9131 subjects. It has an average of 362.6 images for each subject [20]. Only the training portion of VGGFace2 was used to train the FaceNet implementation [14]. More than three million images compose the training portion of the VGGFace2 dataset. The characteristics of VGGFace2 are what initially inspired our use of young and old parent images without any special treatment. We assumed that the final model would be robust to large age differences and gender variation because the VGGFace2 has a high variation on these aspects.

Because the data limitation of kinship datasets and the necessity of data for deep learning methods, we had to use a dataset with a different main purpose than the one of this research, VGGFace2 is used to train and validate FaceNet to extract facial features. FaceNet and VGGFace2 will allow us to leverage the superiority of deep learning models stated by [2] on the kinship verification task.

Labeled Faces on The Wild (LFW) was also used additionally as a test dataset for FaceNet on training. LFW has 1680 classes with two or more distinct photos [21]; these classes are used to test FaceNet performance on every epoch of training. The results obtained on these tests are not used to adjust the weights of the network, only to assess the performance of the trained model without bias.

B. The UB Kinface Dataset

UB KinFace dataset is used to perform cross-validation for kinship verification. The dataset is made of 200 groups of images composed by old parents, young parents, and children (Total of 600 images). Most of the pictures of young parents are in grayscale because of the technology available at the time of the photos; there are also other examples of isolated grayscale images. In Fig. 7 examples of pictures from the UB Kinface dataset are exhibited.

The dataset used for cross-validation is composed of 800 image pairs as described in Table III, the dataset is mounted in a balanced way that follows the structure: [positive child-young parent, negative child-young parent, positive child-old parent, negative child-old parent, ...]. This pattern will repeat throughout the 800 available samples to assure that the tests are well balanced between the four types of examples. The negative child-young parent's pairs are composed of non-kin pairs between child and other young parent individuals, the negative child-old parent's pairs are made of non-kin pairs of child and old parent individuals. We did not separate the types of kinship relations because nearly 80% of the relations are father-son relations [4].

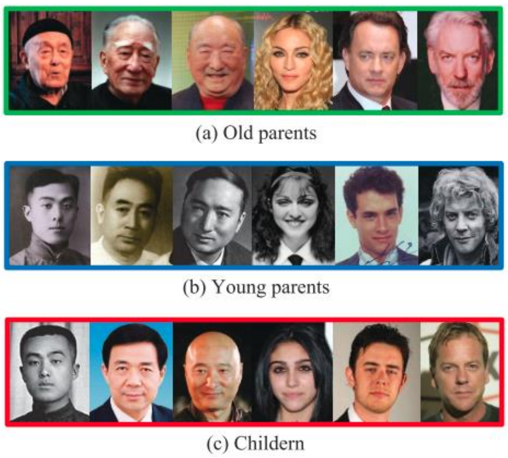


Figure 7. The UB Kinface dataset – source: [1].

TABLE III. STRUCTURE OF THE UB KINFACE DATASET USED FOR CROSS-VALIDATION

Type of relations	Positive	Negative
Child-young parent	200	200
Child-old parent	200	200

C. KinFaceW-I and KinFaceW-II Datasets

The dataset KinFaceW-I is composed of 1066 images distributed as showed in Table IV. The KinFaceW-II is composed by 2000 images, equally distributed between the four types of relations, also present at the KinFaceWI dataset, resulting in 1000 image pairs, 250 for each relation as presented at Table IV. The images are 64x64px already aligned with the algorithm Viola–Jones [4].

TABLE IV. DISTRIBUTION OF IMAGES ON KINFACEW-I AND KINFACEW-II BEFORE FACIAL ALIGNMENT

	KinFaceW-I		KinFaceW-II	
Relation	Pairs	Images	Pairs	Images
Father-daughter	134	268	250	500
Father-son	156	312	250	500
Mother-daughter	127	254	250	500
Mother-son	116	232	250	500

These datasets have less age variation, but they have greater gender variation than the UB Kinface dataset, and it will allow to further assess the performance of the proposed model. Two versions of KinFaceW-I and KinFaceW-II are used one colorful and one grayscale.

The cross-validation datasets for KinFaceW-I and KinFaceW-II are built similarly to the ones constructed with the UB Kinface, adding one positive and negative example of each type of kin relation at a time. The negative example is a combination of a false relation between two types of faces, a father-daughter relation for instance: the image of a father and the image of a non-related daughter is used as a negative example, and it will be put right after a positive example of the same relation. The cycles will be mounted on the most balanced way possible, that means that only one or two cross-validation cycles will be slightly unbalanced.

IV. DEEP LINEAR METRIC LEARNING - DLML

Previous methods have trained and evaluated their solutions on child-young parent pairs and child-old parent pairs separately at the UB Kinface dataset, and on each relation present at KinFaceW-I and KinFaceW-II. We did the contrary, making the cross-validation with one whole dataset at a time.

A. Multi-Task Cascaded Convolutional Network (MTCNN)

We used an MTCNN implementation [14] to perform facial alignment because it provides good performance on hard examples like various poses, illuminations, and occlusions [12]. The MTCNN architecture first resizes the image to different scales to build an image pyramid, which will be the input of a three-phase cascade framework [12]. The framework is described as follow.

- **Proposal Network (P-Net):** On the first stage, a fully convolutional neural network find the candidate facial windows and the bounding box regression vectors. These candidates are found estimating the borders of the face. After that, a non-maximum suppression (NMS) is applied to merge highly overlapped candidates [12].
- **Refine Network (R-Net):** On the second stage, all candidates are processed by another network which discards a significant number of false candidates, executes calibration with bounding box regression, and performs NMS [12].
- **Identify Facial Landmark:** It is similar to the second network, but in this case, the goal is to identify face regions with more supervision, providing five facial landmarks as output [12].

The goal of the research is not performing facial alignment, so, a pre-trained model [14] that uses the weights provided by the authors of the MTCNN paper [12] is used.

The post-MTCNN images will be a stretched version of the face with the size 160x160. The reason to use the stretched face is is that the necessary features for FaceNet are kept after transformation [13], and this allows FaceNet to have the standard input size of 160x160. In Fig. 8 it is possible to see samples of images before and after facial alignment.

Figure 8. Image samples from the UB Kinface database before and after facial alignment with MTCNN.

These aligned face images of the UB Kinface dataset will also be used to create a new dataset that consists of all the images converted to grayscale. This dataset has the purpose of analyzing the impacts of different channel patterns on the results.

B. FaceNet

The FaceNet architecture used in this research has shown one of the best performances on some of the most relevant facial recognition benchmarks like LFW and Youtube Faces DB [13]. Another key factor that inspired the use of FaceNet is the fact that the network generates an array of facial embeddings, assuming the principle that this array can be applied to other purposes, we use it to perform kinship verification.

FaceNet creates an array of 128 dimensions of the face; those dimensions are used on training to create an abstract representation of the face that it is called anchor. The anchor will have a maximum distance of all representations of that face. It is possible to see the FaceNet method in a simple perspective on Fig. 9.

On training, the original paper [13] used a triplet loss function. The training process increases the distance of negative samples and approximates the positive samples, as shown in Fig. 10 [13].

Triplet loss is computationally more costly than training as a softmax classifier using cross-entropy loss and training as a classifier can still offer good results [22]. On this research, Facenet was trained as a softmax classifier.

On the original FaceNet paper, the architecture used was a non-ResNet version of the inception architecture [13]. In this research, the Inception-ResNet-v1 architecture is used because it provides better performance and convergence [23]. The FaceNet implementation used is based on the NN3 architecture of the original paper [13]. This network has input size of 160x160.

To perform testing on the LFW dataset on every epoch of training, the "Pair Matching" protocol with "Unrestricted, with labeled outside data" provided by [24] is used. The 1680 classes with more than two images are used to form pairs of images without overlapping. These pairs will test the distance between the two embeddings created by the network. A class with four images, for instance, will have two pairs of images to evaluate, [0, 1] and [2, 3]. This distance is calculated using the euclidean distance between the two embeddings (L2 norm) as described on (1), with p as one of the embeddings and q as the other.

$$\mathrm{L2} = \sqrt{\sum_{I=0}^{127} (p-q)^2} \tag{1}$$

The test results on LFW during training are not used to adjust the network weights, only to assess the performance of FaceNet.

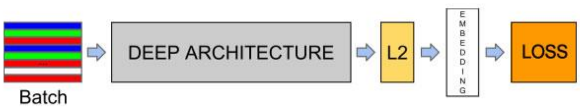


Figure 9. FaceNet original model structure [13].

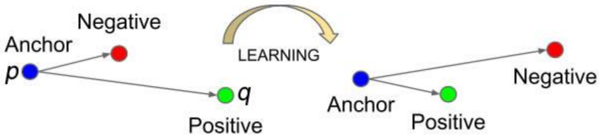


Figure 10. Anchors on the training process [13].

C. Linear Metric Learning for Kinship Verification

The last stage tackles the fact that facial features of similar people lie in a close neighborhood, but this does not necessarily mean that these two people are kin, and the contrary is also true. Enters the metric learning approach that tries to learn what are the right feature differences to detect kin and a non-kin people.

The extracted features of two images are subtracted forming positive difference pairs like [[1, 201], [2, 402], ...], and negatives like [[1, 225], [2, 561]]. Considering 1 to 200 as children, 201 to 400 as young parents, and 401 to 600 as old parents.

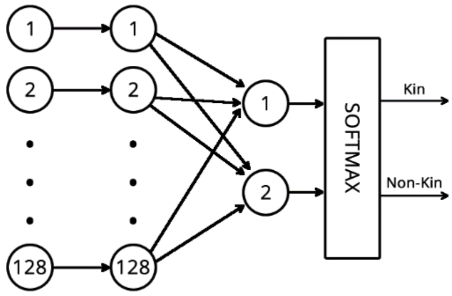


Figure 11. The linear model that receives the non-negative difference array.

The non-negative result of the subtraction is fed into the linear model of Fig. 11 that it will perform the kinship verification. This model has 128 inputs (same size as the embeddings provided by FaceNet). The first layer has the size of 128x1 with bias unities, and it uses the Leaky Rectified Linear Unit (Leaky ReLu) activation function [25]. Next, a fully connected layer with only two outputs finalizes the model, also using the Leaky ReLU activation function. Finally, we use a softmax function to perform the boolean prediction of kin or non-kin. The parameters used for training of the network are displayed in Table V.

On training, dropout is applied after the first layer, the cross-entropy loss function defined on Eq. 2 is used being y_i the predicted value provided by the model, and y_i^l as the expected value. The classical standard backpropagation algorithm performs optimization of the network during training.

$$-\sum y_i^l \cdot \log(y_i) \tag{2}$$

V. Experiments and Results

This section will explore the experiments made in this research and show the results of these experiments.

A. Face Alignment

Facial alignment is performed using MTCNN for five datasets: VGGFace2 for FaceNet training, LFW for FaceNet testing, UB Kinface, KinFaceW-I, and KinFaceW-II for cross-validation kinship verification. The performance of the MTCNN pre-trained model [14] on the five datasets is exhibited on Table V, the before column shows how many images were available before facial alignment, the after shows how many were successfully aligned, and the performance is calculated by comparing how many of the images were processed successfully.

On Table V, it is possible to observe that the low resolution (64x64) facial images already aligned with Viola-Jones algorithm yield the worst performance from MTCNN.

TABLE V. PERFORMANCE OF MTCNN

Relation	Before	After	Performance
VGGFace2	3,141,890	3,138,862	99.90%
LFW	13,233	13,233	100%
UB Kinface	600	600	100%
KinFaceW-I	1066	1045	98.03%
KinFaceW-II	2000	1916	95.80%

B. Feature Extraction: Training, Validation, and Testing

FaceNet is trained with a total of 500 epochs, each epoch has 1000 batches, and each batch has 40 images. On Fig. 12, Fig. 13 and Fig. 14, the 500.0k value on X-axis points the number of batches (500 · 1000).

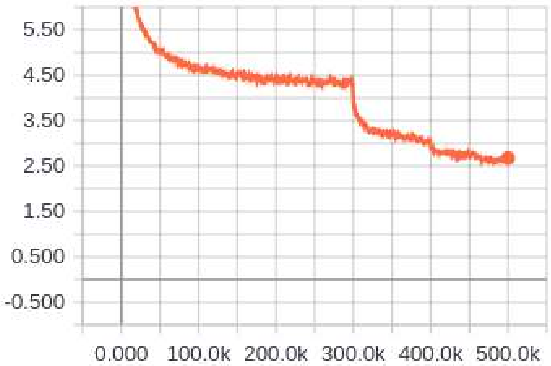


Figure 12. Cross-entropy on training using VGGFace2

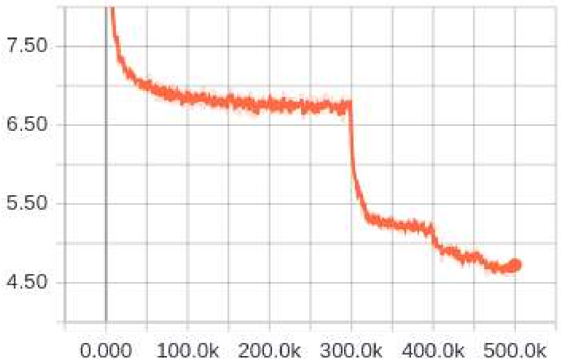


Figure 13. Total loss on training using VGGFace2

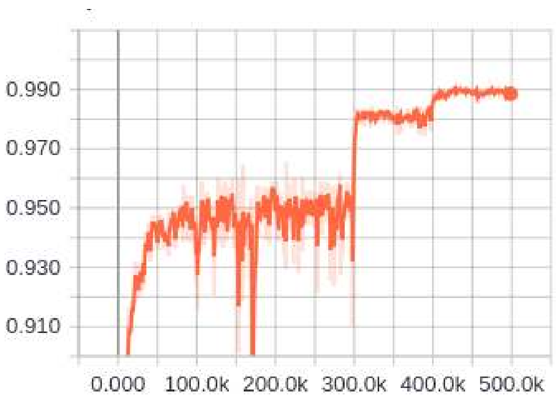


Figure 14. Accuracy on LFW every five epochs

A portion of 0.01% of the VGGFace2 is used for validation on training to calculate the loss and adjust the weights using the Adam optimizer [26]. The cross-entropy loss value of every batch can be observed in Fig. 12. The total sum of the cross-entropy loss can be seen in Fig. 13.

The accuracy on LFW exhibited in Fig. 14 is calculated every five epochs using the euclidean distance. To improve performance and avoid overfitting the fixed image standardization technic and dropout are used. Table VI displays the empirical learning rate used for training with the Adam optimizer.

TABLE VI. EMPIRICAL LEARNING RATE FOR FACENET TRAINING

Epoch	Learning Rate
0-99	0.1
100-299	0.05
300-399	0.005
400-499	0.0005

After completing training, we rerun tests on LFW using the euclidean distance; the accuracy was 98.83%.

C. UB Kinface Cross-Validation on the Linear Metric Learning Model

To train and evaluate the linear metric learning model at the UB Kinface dataset, 800 pairs of images aligned by MTCNN, are used with the five-fold cross-validation protocol and leave-one-out protocol. The experiment results are exhibited in Table VII and Table VIII, the exhibited values are the average of all the values obtained during the cross-validation cycles.

TABLE VII. UB KINFACE 5-FOLD CROSS-VALIDATION RESULTS

Five-fold			
	Accuracy	Precision	Recall
Original color images	41.63%	38.93%	42.75%
Grayscale images	67.38%	65.06%	76.50%

TABLE VIII. UB KINFACE LEAVE-ONE-OUT RESULTS

Leave-one-out			
	Accuracy	Precision	Recall
Original color images	46.62%	44.34%	42.00%
Grayscale images	71.50%	68.01%	82.50%

D. Additional Cross-Validation Experiments with KinFaceW-I and KinFaceW-II

To further assess the performance of the proposed method the method was also tested at two additional datasets, the KinFaceW-I and KinFaceW-II [4]. These two datasets do not have large age differences, but they will assess the proposed method with gender variation, and also low image quality because of the low resolution of the provided images (64x64).

Because one of the main objectives of the proposed method is to provide a more practical solution, the evaluations at the datasets KinFaceW-I and KinFaceW-II are not made on each relation separately; This approach will also allow evaluating the model on gender variation because these two datasets have higher gender variety [4] than the UB Kinface [1].

After performing facial alignment on KinFaceW-I and KinFaceW-II there are images that do not have pairs, these images are removed before the cross-validation dataset is created, this will implicate in two datasets with the number of images shown at Table IX. The cross-validation datasets for KinFaceW-I and KinFaceW-II are built using all the images presented at Table IX in the most balanced possible way.

TABLE IX. IMAGES ON KINFACEW-I AND KINFACEW-II AFTER FACIAL ALIGNMENT AND REMOVAL OF IMAGES WITHOUT PAIRS

	KinFaceW-I		KinFaceW-II	
Relation	Pairs	Images	Pairs	Images
Father-daughter	127	254	234	468
Father-son	148	296	231	462
Mother-daughter	122	244	225	450
Mother-son	113	226	224	448

The linear metric learning model is evaluated on two versions of KinFaceW-I and KinFaceW-II (colorful and grayscale), the cross-validation results are presented at Table X and Table XI.

TABLE X. FIVE-FOLD CROSS-VALIDATION RESULTS ON KINFACEW-I AND KINFACEW-II

Five-fold KinFaceW-I results			
	Accuracy	Precision	Recall
Colorful images	70.49%	67.00%	83.14%
Grayscale images	70.20%	66.18%	83.53%
Five-fold KinFaceW-II results			
	Accuracy	Precision	Recall
Colorful images	68.14%	69.84%	65.57%
Grayscale images	66.72%	65.00%	72.68%

TABLE XI. LEAVE-ONE-OUT CROSS-VALIDATION RESULTS ON KINFACEW-I AND KINFACEW-II

Leave-one-out KinFaceW-I results			
	Accuracy	Precision	Recall
Colorful images	69.90%	65.68%	83.12%
Grayscale images	69.22%	66.37%	78.82%
Leave-one-out KinFaceW-II results			
	Accuracy	Precision	Recall
Colorful images	66.50%	68.87%	61.42%
Grayscale images	66.78%	66.78%	60.56%

The results presented by Table X and Table XI showed that our method ranks better than the human baseline presented at Table II that uses the same images used to train the model. Our proposed method also ranks better than the MNRML [4] and PDFL [16] methods at the KinFaceW-I dataset despite being tested in a non-context-aware manner.

VI. CONCLUSIONS

Our results showed that on the UB Kinface database our Deep Linear Metric Learning method performs fairly well on solving the kinship verification problem, even when there are large age differences, increasing the applicability of the model in a real-world environment. The proposed method shows robustness to the mix of old and newest image data present in the database. The presented method performs well directly on kinship verification with large age variations, without the need for retraining separately for relations on child-young parents and child-old parents as seen in other approaches.

By comparing the UB Kinface results between the original color images and the grayscale images on Table VII and Table VIII, it is possible to verify that even though the features are extracted with a network that it is trained with colorful facial images, the difference of the extracted features provided by FaceNet, when dealing with pair of images in color and grayscale, decreases the performance of our linear model because the distance created by different color channel patterns impacts how expressive the features are to our linear model. This difference led to a worse performance on the images with the original color (colorful and grayscale mixed) than when we converted all images to grayscale.

Despite FaceNet being trained with colorful images, it provides good feature extraction for grayscale images of the UB Kinface database, since these features allowed the linear metric learning stage to achieve good performance with grayscale images.

Comparing the achieved UB Kinface results on Table VII and Table VIII with the results of other methods on Table I, it is possible to observe that our proposed DLML method has very similar accuracy to the fourth best method PDFL with 5-fold cross-validation, 67.30% against 67.38% of our method. With the leave-one-out protocol, our method ranks as the best performance with 71.50% of accuracy. Our DLML method is also superior to the human baseline performance of 56.00%; These results showed that our Deep Linear Metric Learning approach can solve kinship verification with large age variations without tackling separately large age differences.

The proposed DLML method was also able to offer on KinFaceW-I and KinFaceW-II accuracy better than the human baseline, despite not being aware of the relation it was classifying and being trained to classify not one type, but all four types of relations present at the dataset. The consistent lower results on grayscale images at Table X and Table XI also showed that most of the necessary features to identify kinship relations are kept on grayscale images, but working only with colorful images it is possible to achieve better performance.

Finally, by discarding the necessity of detecting and treating large age differences and different types of kin relations our method offers an enhanced all-in-one solution to the kinship verification problem.

VII. Future Work

Our proposed solution showed promising results on the dataset for kinship verification with a large age variation. Further and larger datasets will continue to become available, and for sure further tests would be interesting, especially in order to try to evaluate mother and father's different influences on facial inherited features.

Explore and develop other training methods to extract different features and possibly combine layers for evaluating performance on different subset problems.

References

[1] M. Shao, S. Xia, and Y. Fu, "Genealogical face recognition based on UB KinFace database," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2011, pp. 60–65.

[2] M. Georgopoulos, Y. Panagakis, and M. Pantic. Modelling of facial aging and kinship: A survey. [Online]. Available: http://arxiv.org/abs/1802.04636

[3] N. Kohli, M. Vatsa, R. Singh, A. Noore, and A. Majumdar, "Hierarchical representation learning for kinship verification," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 289–302, 2017.

[4] 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, 2014.

[5] H. Yan, J. Lu, W. Deng, and X. Zhou, "Discriminative multimetric learning for kinship verification," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 7, pp. 1169–1178, 2014.

[6] A. Dehghan, E. G. Ortiz, R. Villegas, and M. Shah, "Who do i look like? Determining parent-offspring resemblance via gated autoencoders," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1757–1764.

[7] Meghan Markle and mom image. [Online]. Available: https://goo.gl/qsgRFU

[8] Joanna Gaines image. [Online]. Available: https://goo.gl/6tnHkN

[9] Prince Charles image. [Online]. Available: https://goo.gl/wAhHv8

[10] Ed Sheeran and prince Charles image. [Online]. Available: https://goo.gl/AE3E4d

[11] Jenifer Lawrence image. [Online]. Available: https://goo.gl/fpU3Cj

[12] K. Zhang, Z. Zhang, Z. Li, S. Member, Y. Qiao, and S. Member, "Joint face detection and alignment using multi - task cascaded convolutional networks," *Spl*, no. 1, pp. 1–5, 2016.

[13] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015.

[14] D. Sandberg. Face recognition using tensorflow. [Online]. Available: https://github.com/davidsandberg/facenet

[15] S. Xia, "Toward kinship verification using visual attributes," in *Proc. International Conference on Pattern Recognition*, 2012, pp. 549–552.

[16] H. Yan, J. Lu, and X. Zhou, "Prototype-Based discriminative feature learning for kinship verification," *IEEE Transactions on Cybernetics*, vol. 45, no. 11, pp. 2535–2545, 2015.

[17] S. Xia, M. Shao, and Y. Fu, "Kinship verification through transfer learning," in *Proc. IJCAI International Joint Conference on Artificial Intelligence*, 2011, pp. 2539–2544.

[18] N. Kohli, R. Singh, and M. Vatsa, "Self-similarity representation of Weber faces for kinship classification," in *Proc. IEEE 5th International Conference on Biometrics: Theory, Applications and Systems*, 2012, pp. 245–250.

[19] A. Chergui, S. Ouchtati, and F. Bougourzi, "LPQ and LDP descriptors with ML representation for kinship verification," in *Proc. International Workshop on Signal Processing Applied to Rotating Machinery Diagnostics*, April 2018.

[20] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. VGGFace2: A dataset for recognising faces across pose and age. [Online]. Available: http://arxiv.org/abs/1710.08092

[21] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, "Labeled faces in the wild: A survey," *Advances in Face Detection and Facial Image Analysis*, pp. 189–248, 2016.

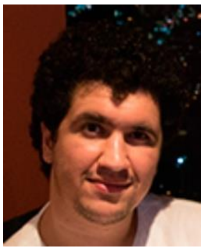
[22] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. British Machine Vision Conference*, 2015, pp. 41.1–41.12.

[23] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. Inception-v4, inception-ResNet and the impact of residual connections on learning. [Online]. Available: http://arxiv.org/abs/1602.07261

[24] G. B. Huang and E. Learned-miller, "Labeled faces in the wild: Updates and new reporting procedures," University of Massachusetts Amherst Technical Report, pp. 14–003, 2014.

[25] B. Xu, N. Wang, T. Chen, and M. Li. Empirical evaluation of rectified activations in convolutional network. [Online]. Available: http://arxiv.org/abs/1505.00853

[26] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. International Conference on Learning Representations*, 2015, pp. 1–15.



Diego Lelis was born in Belo Horizonte, Brazil, in 1988. He received the B.S degree in Computer Science from the University Center of Belo Horizonte (Uni-BH) in 2015. He is currently pursuing the M.Sc. degree on Mechatronics Systems at the University of Brasilia. His research interests include Facial Recognition, Pattern Recognition, Natural Language Processing, Artificial Intelligence, Robotics, and Distributed Systems.

Dibio L. Borges holds a B. Eng. degree in Electrical Engineering from the University of Brasília (1986), an M.Sc. in Computer Science from the University of Brasília (1991) and a Ph.D. from the University of Edinburgh, Scotland (1996), for work in Computer Vision and Artificial Intelligence. He is currently an Associate Professor of the Department of Computer Science at the University of Brasilia, Brazil. He has experience in the areas of Computer Science, Electrical, Computer and Mechatronics Engineering, with research interests in Artificial Intelligence, Computer Vision and Intelligent Sensory Systems.