Explainable Kinship: A Broader View on the Importance of Facial Features in Kinship Recognition

Britt van Leeuwen[a], Arwin Gansekoele[b], Joris Pries[c], Etienne van de Bijl[d] and Jan Klein[e]

Centrum Wiskunde & Informatica, Stochastics group
Science Park 123, Amsterdam, the Netherlands
Email: [a]britt.van.leeuwen@cwi.nl, [b]arwin.gansekoele@cwi.nl,
[c]jorispries@gmail.com, [d]etienne.van.de.bijl@cwi.nl,
[e]jan_g_klein@outlook.com

***Abstract*—Kinship Recognition, the ability to distinguish between close genetic kin and non-kin, could be of great help in society and safety matters. Previous studies on *human* kinship recognition found interesting insights when looking for the most important features. Results showed that analyzing only the top half of a face gives equal or even better performance compared to analyzing the whole face. In this paper, we aim to find the important features for *automated* kinship recognition based on the theory of *human* kinship recognition; this set of features was researched using features from pre-trained metrics from the StyleGAN2 model. Three different experiments were performed focusing on different aspects of facial features. We found that the most important facial features from the selection of 40 features are mostly focused on the facial hair traits. Furthermore, age-related features were found to be very important. This set of features does not entirely comply with the set of features important in *human* kinship recognition. Previous research has shown *human* kinship recognition performance does not decrease when removing the bottom half of the image of the face. In contrast, our results show that for *automated* kinship recognition, removing either the bottom or the top half of a face results in a decrease in the performance of our classifiers. Moreover, only using a selection of facial features corresponding with the important features in human kinship recognition did not prove to be sufficient for the task of Kinship Recognition.**

***Keywords*—kinship recognition; StyleGAN2; Families-in-the-Wild; feature importance; transfer learning.**

I. Introduction

This work is an extension on our previous research in [1] on the importance of facial features in Kinship Recognition.

A. Kinship Recognition

One of the fields in artificial intelligence that is currently of great interest is computer vision. Computer vision is defined as the study domain that revolves around techniques developed to automate seeing and understanding the contents of digital images such as photographs and videos by computers [2]. This new field started to emerge around the 1960s [3]. However, projects such as getting a computer to describe what it saw via a linked camera, proved much more complex than first thought [4]. Computer vision began to rise after a couple of decades, as the internet advanced and, therefore, access to data improved. At that time (the 80s and 90s), it was facial recognition that grew to be more promising. Subsequently, with the boost of the internet and following later social media, we came as far as Facebook using face recognition every day [5]. One of the subjects that keeps computer vision attractive today is face recognition. From unlocking your phone using your face to automated passport checking at an airport, face recognition is used more often than we realize.

The subject of kinship recognition (KR) is fairly new within face recognition. Kinship recognition is the ability to distinguish between close genetic kin and non-kin. The distinction involves people who are directly related and people who are not. One example of the usage of KR is of families who are spread throughout multiple refugee camps. One of these cases involved a father and his daughter being in one camp, while his wife and other children were in another camp. It took them over a year to get reunited by the Red Cross Restoring Family Links [6]. Even with an organization on these reunion cases, it still takes them a considerable amount of time to solve the problem. If a KR system were able to pick them out as a possible match for a kinship relation, it could be of great help. Upon request, a picture could be taken of a refugee and put in a database. This database could then check for kinship relations with other refugees in the same database. If a missing person is registered in a camp, a kinship relation could be detected instantly this way. They would be reunited much faster and more efficiently. Issues such as communication and limited manpower could also be reduced with the discussed automation.

Another example of the usage of KR is focused on safety measures. Imagine a situation where a terrorist is not in the system. No information can be found on them, only an image of their face is accessible. KR systems could try to match possible family members that are in the system of known suspects. This could lead to finding the terrorist sooner or to finding their accomplices. Like this, there are more situations where automated KR would be really helpful. With the reality of these problems, KR can bring families together and provide more safety.

The main contribution of this paper is to make a first step towards understanding automated KR and the importance of facial features in it. In the field of KR, there is a lot of room for improvement, especially on the importance of facial features.

In our research, we tackled whether kinship is recognizable by using a set of extracted facial features with the use of machine learning. Specifically, we focus on what specific set of features is important for automated KR and if this set of features complies with the set of features important in human KR. First presented in this paper is a literature discussion on human as well as automated KR. We then discuss the data in Section II. In Section III, an overview of the used models is presented. Next, we discuss the results of different experiments in Section IV. Lastly, a discussion and conclusion of the presented experiments are given in Sections V and VI.

B. Related work

1) Kinship Versus Look-alike: In various researches, which we will discuss later, it has been shown that automated KR is possible to a certain degree. The main question we are left with is how we would separate the classification of two family members from two people that look alike [7], [8]. On one side, two people could be unrelated but their faces as a whole could look alike. If their facial features would be extracted and compared, the features would probably not have high similarity [9]. They have lower inter-class variations. Inter-personal variations refer to the differences in race or genetics. This includes variations such as eye color and the shape of a nose, features that are not possible to be (easily) changed. On the other side, intra-personal variations refer to variations in features that are easily changed, such as hair, facial accessories, cosmetics, pose and illumination [8]. Their face looks similar due to these intra-personal variations. With this information, we expect each feature separately to not show significant similarities. For example, having a close look at the shape of their nose, it is considerably different from the shape of the other person's nose. The intra-class similarity is higher and the inter-class variation is lower for the two individuals of this example [8].

On the other side, there are two people that are related, a daughter and father for example. They do not necessarily look alike since they differ in age and sex. They are in general not seen as lookalikes. However, when looking at their facial features, most of the time you would see that there is a high similarity for some features, whereas other features would not be similar at all [10]. An example of this is a father and daughter who have a nose and mouth that are very similar. However, their faces as a whole do not look alike, because the rest of their facial features are not similar at all. This would be due to the heredity in kinship, as not all traits inherited from a parent to a child are reflected in the child's appearance.

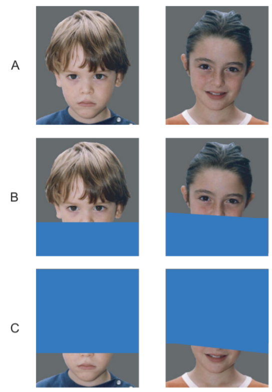
2) Human Kinship Recognition: Studies on human KR contribute to our search for the set of important features in automated KR. Several studies [11]–[13] have been conducted on human KR, which showed that kinship is indeed recognizable by humans. Robinson et al. [14] used the Families-In-the-Wild (FIW) data set for their human performance measurement. This data set contains images of people's faces that are extracted from family pictures. In total, the data set consists of $656,954$ pairs of images that show a kinship relation. Robinson et al. state that humans scored an overall average of 56.6% accuracy. In this experiment, two pictures were shown and a binary classification was performed between related by kinship and unrelated. Other research on KR [11], [12], [15], [16] shows similar results. The results from Lu et al. [14] are shown in Table I. They performed two different experiments to test human KR. The test group is split up into group A and B. Group A was only shown a cropped face region, whereas group B was shown the whole original color images. Group A intends to test kinship verification purely based on face, while group B intends to test kinship verification based on multiple cues including face, hair, skin color, and background [14]. Their results show that the average accuracy of human KR is higher when face, hair color, and background are taken into account compared to when the focus is purely on the face.

TABLE I. RESULTS OF THE TWO EXPERIMENTS ON HUMAN KINSHIP RECOGNITION FOCUSED ON FOUR KINSHIP TYPES BY LU ET AL. [14]. THE NUMBERS REPRESENT THE ACCURACY AND THE DIFFERENT COLUMNS REPRESENT THE KINSHIP RELATIONS FATHER AND SON (F-S), FATHER AND DAUGHTER (F-D), MOTHER AND SON (M-S), AND MOTHER AND DAUGHTER (M-D).

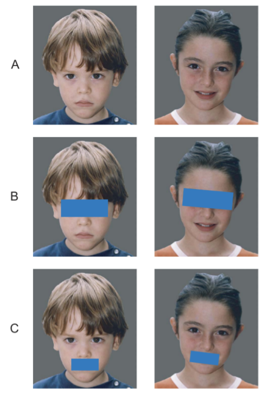
Method	F-S	F-D	M-S	M-D	Mean
HumanA	61.00	58.00	66.00	70.00	63.75
HumanB	67.00	65.00	75.00	77.00	71.00

We take a look at the Feature Importance (FI) in some of these studies on human kinship detection. The reason behind this specific set of features for human KR might be of help in automated KR. One of the studies is by Martello and Maloney [11], [12], who raised the question which parts of a face are most important for human KR. In [11], a study is conducted in which humans were tested on their KR skills based on three separate conditions: (1) the right hemi-face masked, (2) the left hemi-face masked, and (3) the face fully visible. Most interestingly, the results showed that there is no significant difference in results for recognizing kinship by humans when the left or right part of the face is covered. On the contrary, a similar study [12] showed that the covering of the top or bottom part of a face does give a significant difference. The effect on kin recognition performance of masks that covered the upper half or the lower half of the face (experiment 1) and the eye region or the mouth region (experiment 2) were measured. An example of the covering up of facial parts for experiments 1 and 2 can be seen in Figures 1a and 1b below.

In these experiments, it was found that masking the eye region led to a 20% reduction in performance whereas masking the mouth region did not yield a significant difference in performance. This leads us to consider the theory that the performance in KR is dependent on only the upper half of a person's face. Curious is to see how this theory could be used in automated KR. Another discovery is that the eye region contains only slightly more information about kinship than the upper half of the face outside of the eye region. Moreover, the theory is discussed that splitting up face images in different patches can improve the ability of humans to recognize kinship. This would be caused by the mouth area



(a) Experiment 1: Masking the bottom and top half of the face



(b) Experiment 2: Masking the eye area and lip area of the face

Figure 1. Illustration of the masking of faces in [12].

(i.e., the bottom half of the face) containing lesser kinship features as it is subjected to considerable changes during development [17]. The lower face does not reach its final form until early adulthood [18]. Consequently, this area has fewer stable cues to relatedness. Another view discussed [17] is that environmental effects have little influence on the detection of kinship using facial similarities. This indicates that genetically irrelevant facial information is ignored when human KR is performed.

Overall, the theory that we researched is based on the change in performance when using a specific set of facial features compared to facial features from the whole face. The theory implies that there is a mere necessity of these facial features for KR. These are the features that are located in the upper half of the face, which could lead to only requiring specific parts of faces to identify kinship relations. This could lead to more accessible data since only parts of faces are also sufficient for extracting the important features. Moreover, it could decrease the computational cost of KR models.

3) Automated Kinship Recognition: For *automated* KR, several approaches have been proposed. Most approaches are not only focused on machine learning models, but also on feature selection. Feature-based methods aim to preserve facial, genetically determined characteristics in the feature descriptors used for the model. These methods identify local facial features such as inconsistencies in an individual's eyes, mouth, nose and skin from the individual's image. Feature-based methods can decrease computational costs and improve model performance. Most of the proposed models and algorithms were only trained on small data sets.

These are data sets like KinFaceW [14], [19] where only four types of kinship relations were given on a handmade data set of around 150 images [20]. Another data set in the field of KR is TSKinFace (Tri-Subjects Kinship Face Database). This data set has been used in some studies [21], [22], but also proved to be too small. These data sets demonstrated to be insufficient for the task at hand. Most of the proposed classifiers are lower-level models and algorithms which use handcrafted feature extraction (features using information presented in the image itself), Support Vector Machines or K-Nearest Neighbor classifier.

Since 2016, a more extensive data set has been constructed in [13]: Families-in-the-Wild (FIW). This data set has been produced to verify kinship and classify relations [23]. The creators of this data set specify promising results in detecting kinship. Robinson et al. [14] state the best results were obtained when using the SphereFace model with an average accuracy of 69.18% and standard deviation of 3.68. All models performed well compared to previous work, although much improvement could still be made.

After publishing the FIW data set, more research in the field of KR models was done. Many models in KR include the use of FaceNet or other small feature selections for their models' input [24]. FaceNet is a neural network that extracts features of an image. The model provides a mapping from a picture of a face to the Euclidean space. The distances in this space correlate to the amplitude of face resemblance [25]. It produces an output vector to be used as input for a classification model. FaceNet creates embeddings by learning the mapping from images. A disadvantage of using FaceNet is that especially when looking at FI, information gets lost due to lack of feature interpretation [26]. FaceNet could help improve KR models, although we are interested in the similarities between faces by using facial features instead of the faces as a whole. Hence, we use a different approach than FaceNet.

Fang et al. [27] proposed different feature extractions. They performed classification on a pair of images based on the difference between feature vectors of the pair. These pairs are a potential parent and child pair. The top selected features showed to be right eye RGB color, skin gray value, left eye RGB color, nose-to-mouth vertical distance, eye-to-nose horizontal distance and left eye gray value. The results show a high importance for eye related features. 10 out of the 14 top features include the eye area. This study does include specific facial features like eye color, still it only included 22 low-level features. It is indeed shown that most of the selected features are in the upper face area, which complies with the hypothesis.

The top selected features are right eye RGB color, skin gray value, left eye RGB color, nose-to-mouth vertical distance, eye-to-nose horizontal distance and left eye gray value. The results show a high importance for eye related features. 10 out of the 14 top features include the eye area. While this study does include specific facial features like eye color, it only included 22 low-level features. It is indeed shown that most of the selected features are in the upper face area, which complies with the insight.

Most studies on the subject focus on either the overall similarities between faces, or on pre-determined facial feature sets. These studies treat KR tasks similar to the task of a standard facial recognition. Guo et al. [28] argue that kinship classification should be treated differently, since trait similarities are measured across age and gender. Additionally, kinship has a combination of traits and familial traits, which are special for each family pair.

Models proposed by researchers in this field are based on an input of just the images with little to no alterations. Although, some research focus on specific facial features by using for example a weighted graph embedding-based metric learning framework [31] or by using sparsity to model the genetic visible features of a face [32].

Another group of researchers thought of combining the StyleGAN2 algorithm with KR [33]. In the task at hand, there is a restriction that family members should be recognized on the basis of physical facial features. However, several mentioned attempts neglect this constraint and do not employ any facial landmark before using a classification model. For this reason, Nguyen et al. [33] experimented with KR models using StyleGAN2 as an encoder to incorporate a facial landmark map. This method resulted in an average accuracy of 0.548 for recognizing kinship. Against expectations, no improvement was shown in the results from using StyleGAN2 in this manner, which is presumed to be due to the lack of a proper classification and thus it is argued to need more investigation. An algorithm proposed by Guo et al. [28] use familial traits extraction and kinship measurement based on a stochastic combination of the familial traits. The authors use a similarity score based on a Bayes decision for each pair of facial parts. However, facial features used by the algorithm are limited to the eyes, nose and mouth and, in line with the observations by Guo et al. [28], more parts of the face could be explored. Existing data sets use faces from the same family picture, so models learn about the background similarity. This causes the models to get a higher performance, but when tested on real life pictures, not taken from a family picture, the performance could be lower. When using pre-determined features, this does not present a problem.

II. Data

We used the Families in The Wild (FIW) data set. FIW is made up of 11,932 natural family photos of 1,000 families. Other data sets contain less images. KinFaceI consists of 1000 pairs of pictures (so even less unique pictures) [29] and TSKinFace consists of 787 pictures [30]. This makes the FIW data set by far the biggest data set being used for KR.

The data contains images of people's faces that are extracted from family pictures, hence the images vary in quality. All images of the persons are of the same size (108x124 pixels). Some pictures are zoomed in on more than others, which causes this quality difference. In some images, the face and its facial features are clearly shown, but other images are very blurry.

The data is split up into training and test data using hold-out cross-validation. The data is split up in a 70/30 split, respectively. The training set consists of information on families, persons and relations between persons including images of the persons. The data is distributed as follows. An average of about 12 images per family, each with at least 3 and as many as 38 members. Each family is assigned a unique id, each person is assigned an id and each image collected is assigned a unique id. The data set includes good-quality images of a person's face, but also blurry images of faces, as shown in Figures 2a and 2b, respectively.

(a) Image from data set

(b) Blurry image from data set

Figure 2. Example data from the Families-in-the-Wild data set

A file containing all matches in the training data set is available. However, this does not include data on combinations of persons that do not have a familial relationship. So, these pairs have been constructed by taking random pairs of images from the set of training images of the FIW data set. This is excluding existing related pairs and each pair is unique. This resulted in 205,285 related and 205,285 unrelated pairs of images. The kinship relations are labeled as related. Of all related data points, 21% of the data points are zeroth generation (siblings), 75% are first generation (parents and children) and 4% are second generation (grandparents and grandchildren).

Binary classification is used for predicting relatedness, so the data set is balanced accordingly. Since the focus is on the importance of each of the facial features in recognizing kinship, we have a split in the data between related and unrelated. The distinction between the types of kinship relations is not made. For now, the types are not taken into the classification process, considering the aim is to have a general interpretation of the important facial features. However, the distribution between the types of pairs can help to understand the possible patterns found in feature importance.

StyleGAN2 metric: linear separability

This research is focused on FI in KR. To be able to understand the FI of a model, the features extracted from a model should be interpretable. To collect a bunch of features and to avoid having to do manual annotation, we decide to use a feature description method from the StyleGAN2 model. With this, it can be easily deducted which of the features of a face are seen as most important by a model for detecting kinship. The pictures in the data are of size 108x124, while the StyleGAN2 description method expects pictures of size 256x256 as input. Interpolation of the pictures in the data is used to overcome this problem. The StyleGAN2 model contains a certain metric called linear separability. StyleGAN2's linear separability metric can be used to steer a generated picture in a certain direction by specifying 40 facial features which are shown in Table IV. For example, the models can be used to make the generated face have blond hair and high cheekbones. What we are most interested in for this research are the pre-trained models used in StyleGAN2 which produce probabilities of the 40 features to be true for an image of a person.

TABLE II. FACIAL FEATURES OF LINEAR SEPARABILITY METRIC

1) 5-o-clock-shadow,
2) arched eyebrows,
3) attractive,
4) bags under eyes,
5) bald,
6) bangs,
7) big lips,
8) big nose,
9) black hair,
10) blond hair,
11) blurry,
12) brown hair,
13) bushy eyebrows,
14) chubby,
15) double chin,
16) eyeglasses,
17) goatee,
18) gray hair,
19) heavy make up,
20) high cheekbones,
21) male,
22) mouth slightly open,
23) mustache,
24) narrow eyes,
25) no beard,
26) oval face,
27) pale skin,
28) pointy nose,
29) receding hairline,
30) rosy cheeks,
31) sideburns,
32) smiling,
33) straight hair,
34) wavy hair,
35) wearing earrings,
36) wearing hat,
37) wearing lipstick,
38) wearing necklace,
39) wearing necktie,
40) young.

The metric was trained using the CelebA Data set (Celeb-Faces Attributes Data set). This is a face attributes data set with 202,599 celebrity images, each with five landmark locations and 40 attribute annotations. StyleGAN2's linear separability metric is meant to be used for the StyleGAN2 model and its corresponding data. We are interested in using the metric on the data from FIW. The information gathered from the linear separability metric (the facial features) is used as a starting point for the kinship classification models. Transfer learning does not only save time, but it also has the possibility of making a learning process more efficient [34].

Consequently, some adjustments to the data were necessary to apply the metric. This resulted in an output of 40 features for all images in the data set, which then could be used to train the chosen automated KR models. As data points for the models, we chose a list of length 40 and a list of length 80, composed of the metric values for the features per two pictures. Two input types were experimented with: (1) a list of 80 features, consisting of 40 features per image, and (2) a list of 40 features, taking the absolute difference of the feature values between the images per feature.

III. Model description

We implemented and tested several models to see how well the models work on our data and to find a recurring pattern in FI. For all models, the FI is investigated. The results of this are then used to understand whether the theory of human KR will hold for automated KR as well. Various machine-learning models were selected for this task. For each model, the accomplished accuracy is obtained by K-fold cross-validation. The number of folds is set to 10 and the data is shuffled before splitting into batches.

Machine learning methods

Using StyleGAN2's linear separability metric on our data results in an output of 40 features for all images in the data set, which then are used to train the models. As data points for the models, we chose a list of either length 40 or 80, composed of the metric values for the features per two pictures. The five models we decided to experiment with are decision tree, random forest, Gaussian naive Bayes, linear support vector machine and logistic regression.

Decision Tree: First, we have the decision tree algorithm with a maximum depth set to 10, where we obtain the FI by using the Gini importance. The Gini importance is calculated as shown in Equation (1) with ni_j the importance of node j, w_j the weighted number of samples reaching node j, C_j the impurity value of node j and $left(j)$ and $right(j)$ the child nodes from left and right split respectively on node j.

$$ni_j = w_j C_j = w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)} \quad (1)$$

The Gini importance value ni_j for feature i is then used for the feature importance of feature i with Equation (2) where fi_i is the importance of feature i and ni_j the importance of node j. These values were then normalized to a value between 0 and 1 by dividing by the sum of all feature importance values [35].

$$fi_i = \frac{\sum_{\text{j: node j splits on feature i}} ni_j}{\sum_{\text{k}\in \text{ all nodes}} ni_k} \quad (2)$$

Decision trees are easily interpretable. Because of the use of decision-making logic, the information on the model's features is easily extracted in a comprehensible form [36]. Decision trees have a built-in feature selection, which is beneficial for our research [37]. However, overfitting is common when using decision trees. This is due to the trees being too complex.

Random Forest: Second, we have the random forest consisting of 100 trees, where the FI is obtained by using the impurity importance. The feature importance is computed as an average over all trees. The splitting rules of a random forest scale down the impurity presented by a split. When a split shows a considerable decrease in impurity, the split is seen as important. This theory results in the impurity importance calculation for a variable in the random forest as shown in Equation (3) where $RFfi_i$ is the importance of feature i in the random forest, $normfi_{ij}$ the feature importance for feature i in tree j normalized, T_{all} the set of all trees, and T the number of trees in the random forest [35], [38], [39].

$$RFfi_i = \frac{\sum_{j \in T_{all}} normfi_{ij}}{T} \quad (3)$$

Where decision trees are susceptible to overfitting, specifically when a tree is notably deep, the random forest algorithm reduces this possibility of overfitting. This is due to random forest algorithms constructing multiple decision trees where more combinations of conditions are represented [40].

Gaussian Naive Bayes: Then, we have the Gaussian Naive Bayes, which obtains FI by using the permutation importance. The permutation importance is calculated by taking the difference between the prediction error of the baseline metric and the prediction error of the permuted feature metric as shown in Equation (4). The permutation importance is obtained as follows. First, the model m is scored s on data D. Then permutation variable importance of feature j is calculated. For each feature, the feature column is randomly shuffled to create an adjusted version of the data $\hat{D}_{k,j}$. The model is then again scored on the data, although now with the feature f replaced by the adjusted version. This results in the score $s_{k,j}$, after which the importance i_j for feature f_j can be calculated with Equation (4) [41].

$$i_j = s - \frac{1}{K}\Sigma_{k=1}^{K} s_{k,j} \tag{4}$$

This algorithm generally works very fast and can easily predict the class of a test data set. It is not sensitive to irrelevant features [42]. The naive Bayes algorithm does perform the best overall when there is independence between the features, while some of our features are dependent. It assumes that all the features are independent [43]. However, even without independence between the features, the naive Bayes algorithm generally performs well.

Linear Support Vector Machine: Next is the linear Support Vector Machine (SVM), where the weights of the model are used to determine FI. These weights are used as vector coordinates. The vector coordinates are orthogonal to the hyperplane represented by the weights. The directions of the vectors represent the class prediction. The difference in the size of the weights is used to determine the feature importance [44]. SVMs have a low risk of overfitting [45], outliers have less influence in the algorithm and the SVM algorithm is relatively memory efficient [46]. Nonetheless, understanding the final SVM model and interpreting the feature importance is difficult. Additionally, SVMs are usually not very suitable for large samples of data, although LSCVs handle this better [47].

Logistic Regression: Lastly, we have logistic regression, where the FI is determined by using the coefficients of the decision function. To get a feeling for the “influence” of a given parameter in a linear classification model, the magnitude of the coefficient for each feature times the standard deviation of the corresponding parameter in the data is considered. The positive coefficients correspond to outcome 1 (related) and the negative coefficients correspond to outcome 0 (unrelated). This means that a higher positive value of the corresponding feature pushes the classification more towards the negative class [48]. Logistic regression is easy to implement and interpret and very efficient to train. Therefore, it does not require high computation power [49]. The algorithm does make the assumption of linearity between the log odds and the independent variables [50].

IV. Results

Three different approaches have been researched, the original StyleGAN2 description method, the pre-selected features method and the bottom and top masked method. The results of these approaches are discussed and an overview of the results is provided.

A. Original StyleGAN2 descriptor experiment

The initial approach is taking the results of the StyleGAN2 model and using them as input for the different algorithms. Over all images, we calculated the probabilities of the image complying with the given 40 features. Extracting 40 features per picture resulted in 80 different values since we were working with two images per data point. The FI was determined per model. For the 80 feature input, we took the sum of each feature per picture. An overview of all the results from the StyleGAN2 descriptor experiment can be found in Table VI and Table VII.

Decision Tree: The accuracy of the decision tree with 40 features as input has a mean of 0.61 with a standard deviation of 0.003. The 80 features input gives a mean accuracy of 0.66 with a standard deviation of 0.005. The model is more leaning towards giving a positive (related) classification. The feature importance for the decision tree model is shown in Figure 3. For the decision tree model with input of 40 features, *arched eyebrows*, *no beard* and *heavy makeup* are the most important features. For the input of 80 features, the top most important features include *young*, *no beard* and *wearing necklace*.

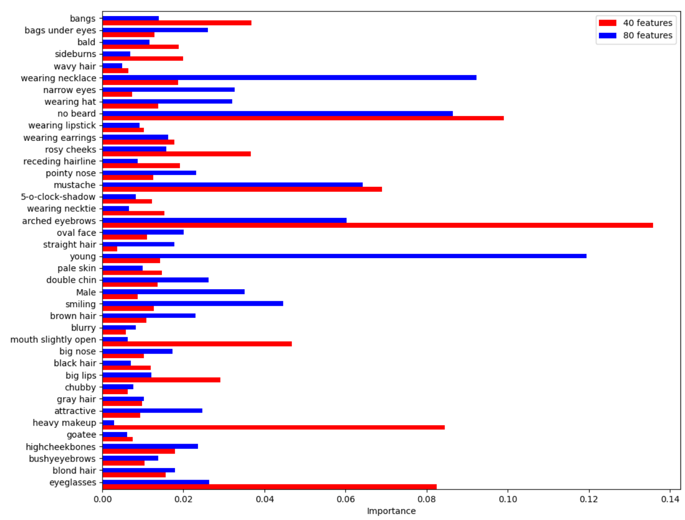


Figure 3. Barplots of feature importance for the decision tree model.

Random Forest: The accuracy of the random forest with 40 features as input has a mean of 0.74 with a standard deviation of 0.003. The 80 features input gives a mean accuracy of 0.80 with a standard deviation of 0.004. The model does not have a clear preference for either a positive or negative classification. With the model giving 51.39% and 50.63% positive classifications for 40 and 80 features respectively, the even distribution of the data in half-positive and half-negative data points is represented well with a slight deviation towards positive classifications. The feature importance for the random forest model is shown in Figure 4. For the random forest model with input of 40 features, *arched eyebrows*, *mustache* and *heavy make up* are the most important features. For the input of 80 features, the top most important features include *young*, *no beard* and *mustache*.

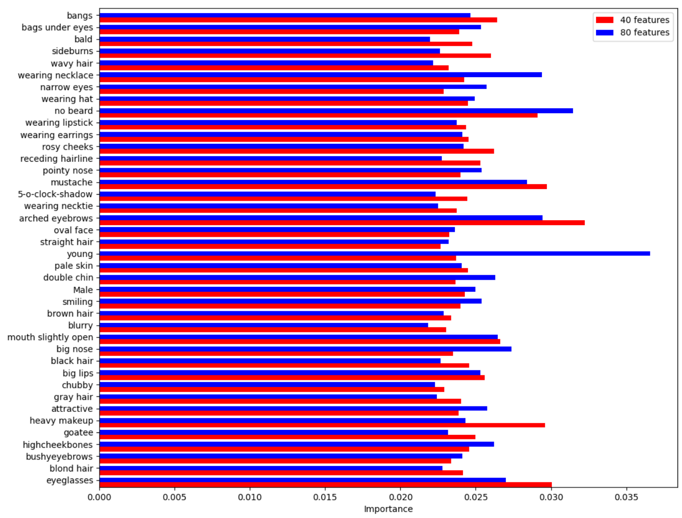


Figure 4. Barplots of feature importance for the random forest model.

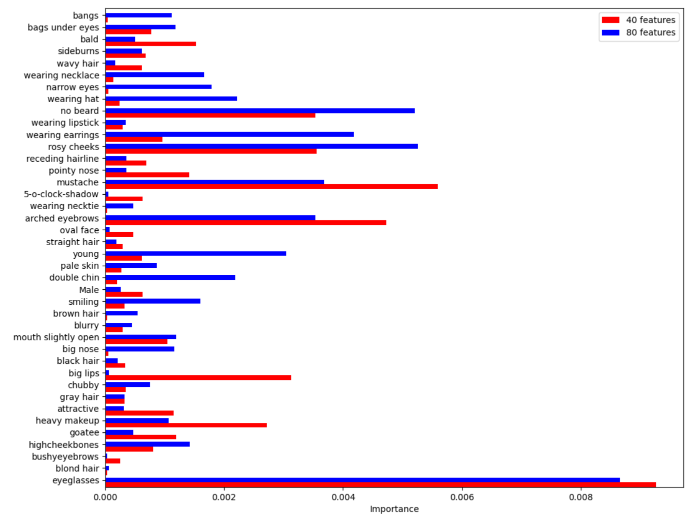


Figure 5. Barplots of feature importance for the naive Bayes model.

Gaussian Naive Bayes: The accuracy of the Gaussian naive Bayes with 40 features as input has a mean of 0.60 with a standard deviation of 0.004. The 80 features input gives a mean accuracy of 0.59 with a standard deviation of 0.005. The model has a preference for positive classification. With the model giving 59.56% and 64.21% positive classifications for 40 and 80 features respectively, most errors are false positives. The feature importance for the Gaussian naive Bayes model is shown in Figure 5. For the Gaussian naive Bayes model with input of 40 features, *eyeglasses*, *mustache* and *arched eyebrows* are the most important features. For the input of 80 features, the top most important features include *eyeglasses*, *rosy cheeks* and *no beard*.

Linear Support Vector Machine: The accuracy of the linear SVM with 40 features as input has a mean of 0.59 with a standard deviation of 0.004. The 80 features input gives a mean accuracy of 0.63 with a standard deviation of 0.005. The model does not have a clear preference for a positive or negative classification. With the model giving 47.08% and 52.92% positive classifications for 40 and 80 features respectively, we see a slight effect of the different input values. The 40 values input gives the model a bit more lenience towards negative classification and the 80 values input gives the model slightly more lenience towards positive classification. The feature importance for the LSVM model is shown in Figure 6. For the LSVM model with input of 40 features, *arched eyebrows*, *no beard* and *heavy make up* are the most important features. *no beard* and *arched eyebrows* are also among the most important features for the input of 80 features. Here the top most important features include *arched eyebrows*, *narrow eyes* and *no beard*.

Logistic Regression: The accuracy of the logistic regression with 40 features as input has mean 0.60 with a standard deviation of 0.003. The 80 features input gives a mean accuracy of 0.63 with a standard deviation of 0.005. The model does not have a clear preference for a positive or negative classification. The model gives 50.40% and 51.61% positive classifications for 40 and 80 features respectively, which shows the balance of the data with a slight deviation towards positive classification. The feature importance for the Logistic Regression model is shown in Figure 7. For the logistic regression model with input of 40 features, *arched eyebrows*, *no beard* and *eyeglasses* are the most important features. These are also among the important features for the input of 80 features. Here the top

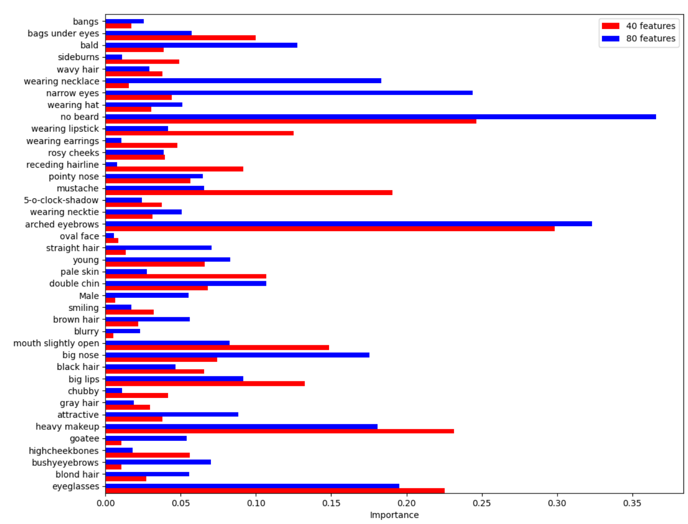


Figure 6. Barplots of feature importance for the LSVM model.

most important features include *no beard*, *arched eyebrows* and *pale skin*.

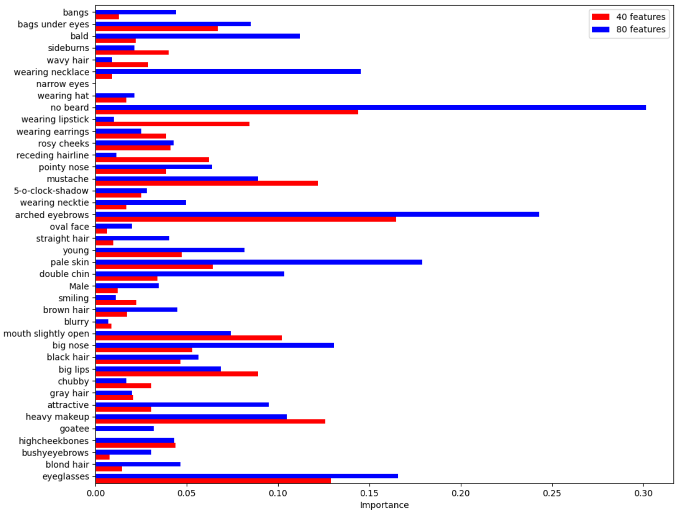


Figure 7. Barplots of feature importance for the logistic regression model.

B. *Selected StyleGAN2 descriptor experiment*

The second approach is based a certain selection of features. Two different selections were experimented on. The first is focused on the human KR theory. A pre-selection of features is applied to the selected models. The selection of features is focused on the top half of a face. This selection is shown below in Table III.

TABLE III. SELECTED SET OF TOP HALF FACIAL FEATURES OF LINEAR SEPARABILITY METRIC

1) Wavy hair,
2) 5-o-clock-shadow,
3) arched eyebrows,
4) bags under eyes,
5) bald,
6) bangs,
7) black hair,
8) blond hair,
9) brown hair,
10) bushy eyebrows,
11) eyeglasses,
12) gray hair,
13) high cheekbones,
14) narrow eyes,
15) receding hairline,
16) sideburns,
17) straight hair,
18) wearing earrings,
19) wearing hat.

This approach, despite the supporting theory, did not give better results than using all the features. The expectation was stability in the accuracy scores by only focusing on the allegedly most important features. However, most models were less successful and only some of the models continued to perform roughly the same. The accuracy scores for the models using the 19 selected features are given in Table V. For this experiment, the differences were taken between the features, so the results should be compared to the results of the method where the input was 40 features as shown in Table V.

The second approach is based on the selection of features found to be most important according to the original Style-GAN2 approach. The top most important features for this approach can be found in Table IV. When using only these features as input, the results are as shown in Table V.

TABLE IV. SET OF MOST IMPORTANT FOUND FACIAL FEATURES OF LINEAR SEPARABILITY METRIC

1) Arched eyebrows,
2) eyeglasses,
3) heavy makeup,
4) mustache
5) narrow eyes,
6) no beard,
7) young.

TABLE V. OVERVIEW OF RESULTS, PRESENTED AS ACCURACY, FOR THE TWO SETS OF SELECTED FEATURES COMPARED TO THE RESULTS OF ALL FEATURES

	Selection Top	**Selection Important**	**All 40**
Decision Tree	0.61	0.61	0.59
Gaussian Naive Bayes	0.57	0.59	0.61
Support Vector Machine	0.57	0.57	0.60
Logistic Regression	0.57	0.57	0.60
Random Forest	0.71	0.62	0.74

C. *Masked StyleGAN2 descriptor experiment*

To support the theory we found, all of StyleGAN2's linear separability features were taken of not the original image, but over an image with the bottom part of the face masked black like shown in Figure 8. The same was done with the top part of the face masked black, comparable to the experiments performed by Martello et al. [11], [12]. All the models are exactly the same as for the original StyleGAN2 description method. Only the input changed.

(a) (b)

Figure 8. Example data from the Families in the Wild data set with bottom masked (a) and top masked (b)

Bottom half masked: This experiment was done with all models previously used in the original StyleGAN2 descriptor experiment. The accuracy and FI were obtained for the decision tree, random forest, Gaussian naive Bayes, LSVM and logistic regression models. An overview of the accuracy and important features for all the models from the bottom masked StyleGAN2 descriptor experiment can be found in Table VI and Table VII. Again, the results show that the 80 value input gives an overall better performance than the 40 value input and the best-performing model is the random forest for both inputs. Some of the most important features for the bottom masked approach are related to the nose (*pointy nose* and *big nose*) and the hair (*grey hair*, *blond hair* and *waivy hair*).

Top half masked: This experiment was done with all models previously used in the original StyleGAN2 descriptor experiment. The accuracy and FI were obtained for the decision tree, random forest, Gaussian naive Bayes, SVM and logistic regression models. An overview of the accuracy and important features for all the models from the bottom masked StyleGAN2 descriptor experiment can be found in Table VI and Table VII. Again, the results show the 80 value input gives overall better performance than the 40 value input and the best performing model is the random forest for both inputs.

TABLE VI. ACCURACY FOR THE 40 AND 80 VALUE INPUT PER EXPERIMENT: COMPLETE, BOTTOM MASKED AND TOP MASKED

	40 Compl.	**40 Bottom**	**40 Top**	**80 Compl.**	**80 Bottom**	**80 Top**
Decision Tree	0.61 ± 0.003	0.57 ± 0.004	0.57 ± 0.003	0.66 ± 0.005	0.64 ± 0.004	0.65 ± 0.003
Random Forest	0.74 ± 0.003	0.62 ± 0.003	0.63 ± 0.002	**0.83 ± 0.004**	0.81 ± 0.001	0.82 ± 0.001
Gaussian Naive Bayes	0.60 ± 0.004	0.53 ± 0.003	0.55 ± 0.002	0.59 ± 0.005	0.55 ± 0.003	0.57 ± 0.002
Support Vector Machine	0.59 ± 0.004	0.55 ± 0.002	0.57 ± 0.002	0.63 ± 0.005	0.60 ± 0.002	0.61 ± 0.002
Logistic Regression	0.60 ± 0.003	0.55 ± 0.003	0.57 ± 0.002	0.63 ± 0.005	0.59 ± 0.003	0.61 ± 0.002

TABLE VII. MOST IMPORTANT FEATURES PER EXPERIMENT

	Complete	**Bottom Masked**	**Top Masked**
Decision Tree	young, no beard, arched eyebrows, eyeglasses	attractive, blond hair, pointy nose, grey hair	young, no beard, arched eyebrows, eyeglasses
Gaussian Naive Bayes	eyeglasses, no beard, young, arched eyebrows	wavy hair, blond hair, pale skin, heavy makeup	eyeglasses, no beard, young, arched eyebrows
Support Vector Machine	young, no beard, pointy nose, arched eyebrows	grey hair, pale skin, wavy hair, big nose	young, no beard, pointy nose, arched eyebrows
Logistic Regression	blurry, no beard, wearing necklace, pointy nose	wavy hair, young, grey hair, big nose	blurry, no beard, wearing necklace, pointy nose
Random Forest	young, no beard, mustache, arched eyebrows	pointy nose, grey hair, smiling, attractive	young, no beard, mustache, arched eyebrows

V. DISCUSSION

Multiple models have been tested on FI. Some approaches were based on the human KR experiments from [11], [12]. These experiments showed a certain area of the face to contain the important facial traits needed for KR. We researched the set of features that is most important for automated KR. Pre-trained metrics from the StyleGAN2 model that are meant to be used for synthesizing artificial examples of faces were used. The pre-trained models give 40 values for specific facial features. These 40 values can also be taken from pictures using the pre-trained models. These values were used as input for our machine learning models: decision tree, random forest, Gaussian naive Bayes, support vector machine and logistic regression. These models were trained and evaluated to show which of the features were seen as most important by the models. More experiments were conducted with the top and bottom parts of a face masked black to also test the theory of human KR.

Major findings: Interesting results were found when comparing the different models using the original StyleGAN2 description method. Four out of five models had a higher accuracy score when all features for both pictures were kept separate. The models are able to learn about combinations of different features between the two pictures, which has a positive influence on the accuracy score of the models.

The best-performing model seems to be the random forest. Since this model has a very high accuracy compared to the other models, we are specifically interested in its corresponding FI scores. Accordingly, we mainly focus on the results of the random forest model. This model gives high importance values to the features *young*, *no beard*, *mustache* and *arched eyebrows*. It is also noticeable that in two of the five models, the feature *young* is found to be very important and in the other three models, the FI increases when using 80 features instead of 40 features as input. On top of that, in all models, the features *arched eyebrows* and *no beard* are in the top four of the most important features for the model. There is a clear pattern in the importance of facial hair. Beards, mustaches and arched eyebrows are found to be important features for most of the models. Another pattern is the age difference. This gives us reason to believe that the combination of facial hair and the age of a person is strongly correlated to the classification. While the correlation scores do not show a correlation between the two features, the combination of the features does matter when comparing two pictures. A reason for these features to be found important is that most of the kinship relations (75%) in the data set are zero-generation and first-generation relations. Young people are not able to grow facial hair, if they have the genes, it comes with age. This would explain why both facial hair and age are found to be more important.

The set of features that were found to be the most important in our research does not comply with the selection of features proposed by Fang et al. [27]. The set of features used in their research is different, although it is clear that the eye area was found to be the most important by them. Contrasting, the set of important features we found is not particularly focused on the eye area.

For the masked experiment, all five models had a higher accuracy score when all features for both pictures were kept separate. When looking at the bottom masked method results, a clear decrease in the performance is found compared to the original StyleGAN2 description method. Remarkable is that the feature *young* and the features on facial hair are not found in the top features of almost all models. The original StyleGAN2 approach showed these features to be very important. This leads to the believe that the bottom part of a face is essential for extracting the feature *age*. This would also explain why the feature *grey hair* is found to be important in three out of five models. Grey hair is usually a sign of a higher age. When looking at the top masked method results, a decrease in the accuracy is found, although this decrease is not as excessive as with the bottom masked method. Above is mentioned that the feature young is likely to be extracted from mostly the bottom of a face. However, this is not shown in the results of the top masked method. It is curious that the feature *young* is still not found to be one of the most important. Like the original approach, the top masked method shows the feature *arched eyebrows* to be important. Although a pattern is difficult to find in the top masked method results.

For the bottom masked approaches the difference with the original approach is clear. Where humans showed equal or even better performance when masking the top half of a picture, the algorithms showed the opposite effect.

The set of features does not comply with the set that we expected it to comply with. The gathered results do not give any information that would validate the hypothesis that the most important features would be in the upper half of the face, specifically the eye region. On the contrary, the results are more lenient towards age and facial hair traits to be of great importance. As for the approach with the pre-selected StyleGAN2 linear separability features, the results showed us no improvement when focusing on solely the upper half of the face. When considering the results of the pre-selected StyleGAN2 and the masked StyleGAN2 descriptor experiments, rejecting the hypothesis is even more reasonable.

Limitations: The data set might not be very compatible with the StyleGAN2 metrics, which is an uncertainty. However, as of now, there are no other data sets that contain enough images which are of adequate quality. So we have to accept this limitation for now. An issue was also encountered when using the linear separability metric for a different purpose than StyleGAN2. The results for the top masked method showed one very noteworthy important feature, namely the *arched eyebrows* feature. This feature should be focused on the top part of a face. However, it is found to be important when the top part of a face is masked. More features which show unusual behavior are *smiling* and *pointy nose*, since these are found to be important when masking the bottom half of the face. This is one of the problems that is encountered when combining StyleGAN2 metrics with other models. The models that are trained for the linear separability metric behave differently than intuitively expected. Using the metric in tasks for which it is not initially intended can cause limitations to the models.

Unexpected findings: A surprising matter is the difference in performance between the top masked and bottom masked StyleGAN2 description method. Masking the bottom half of the face decreased the performance. As masking the top half of the face decreased the performance as well, it still performed better than the bottom masked method. This is against expectations and raises the question of whether the bottom part of a face contains more information than the top part of a face does for KR.

VI. Conclusion

We researched the set of features that is most important for automated KR. For this, multiple models have been tested on FI. The results showed that the most important facial features from the selection of 40 features are mostly focused on the facial hair traits and age-related features.

One of the issues we ran into is on transfer learning. The question rises whether StyleGAN2 is compatible enough for transfer learning when combined with our data set. It could be more effective to write a new metric that focuses on more solid facial features. Despite that, the StyleGAN2 metrics are the most elaborate method for finding pre-determined facial features. Other models do not include as many facial features or need manual annotation. It would be contributory to find a way to annotate all parts of the face for many more features to train the models on.

In conclusion, this paper is an important first step towards understanding automated KR, but there are many challenges to be faced before it can be used in real-world applications. As it is now, a large set of clear pictures of complete faces are needed for a model to perform decently. Learning more about the most important parts of our face for automated KR is the next step to take to improve the field of KR.

VII. Acknowledgment

We thank Rob van der Mei and Sandjai Bhulai for their useful comments on drafts of this paper.

References

[1] B. E. van Leeuwen, A. Gansekoele, J. Pries, E. van de Bijl, and J. Klein, “Explainable Kinship: The Importance of Facial Features in Kinship Recognition”, Iaria Congress 2022 Proceedings, pp. 54-60, 2022

[2] J. Brownlee, “Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python”, https://books.google.nl/books?id=DOamDwAAQBAJ, Machine Learning Mastery, 2019.

[3] R. Szeliski, “Computer Vision: Algorithms and Applications”, https://books.google.nl/books?id=bXzAlkODwa8C, Springer London, 2010.

[4] S. A. Papert, “The Summer Vision Project”, http://hdl.handle.net/1721.1/6125, 1966.

[5] “A brief history of facial recognition - NEC New Zealand”, https://www.nec.co.nz/market-leadership/publications-media/a-brief-history-of-facial-recognition/, May 2020, (Accessed on 21/10/2022).

[6] E. Seselja, “How the Red Cross and a radio reconnected a family torn apart by conflict - ABC news”, https://www.abc.net.au/news/2021-08-29/red-cross-reconnect-family-separated-by-conflict-after-16-years-/100413214, August 2021, (21/10/2022).

[7] F. Schroff, T. Treibitz, D. Kriegman, S. Belongie, “Pose, illumination and expression invariant pairwise face-similarity measure via Doppelgänger list comparison”, International Conference on Computer Vision, pp. 2494-2501, 10.1109/ICCV.2011.6126535, 2011.

[8] H. Lamba, A. Sarkar, M. Vatsa, R. Singh, and A. Noore, “Face recognition for look-alikes: A preliminary study”, International Joint Conference on Biometrics (IJCB), pp. 1-6, 10.1109/IJCB.2011.6117520, 2011.

[9] N. L. Segal, J. L. Graham, and U. Ettinger, “Unrelated look-alikes: Replicated study of personality similarity and qualitative findings on social relatedness”, Personality and Individual Differences, vol. 55(2), pp. 169-174, 2013

[10] G. Guo, X. Wang, “Kinship Measurement on Salient Facial Features”, IEEE Transactions on Instrumentation and Measurement, vol. 61(8), pp. 2322-2325, 2012.

[11] M. F. Dal Martello and L. T. Maloney, “Lateralization of kin recognition signals in the human face”, The Association for Research in Vision and Ophthalmology - Journal of vision, vol. 10(8), 2010.

[12] M. F. Dal Martello and L. T. Maloney, “Where are kin recognition signals in the human face?”, The Association for Research in Vision and Ophthalmology - Journal of vision, vol. 6(12), 2006.

[13] J. P. Robinson, M. Shao, Y. Wu and Y. Fu, “Family in the Wild (FIW): A Large-scale Kinship Recognition Database”, CoRR, abs/1604.02182, 2016.

[14] J. Lu, X. Zhou, Y. Tan, Y. Shang and J. Zhou, “Neighborhood Repulsed Metric Learning for Kinship Verification”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36(2), pp. 331–345, 2014.

[15] L. M. DeBruine, F.G. Smith, B. C. Jones, S. C. Roberts, M. Petrie and T. D. Spector, “Kin recognition signals in adult faces”, Vision research - Elsevier, vol. 49(1), pp. 38–43, 2009.

[16] G. Kaminski, S. Dridi, C. Graff, and E. Gentaz, “Human ability to detect kinship in strangers’ faces: effects of the degree of relatedness”, Proceedings of the Royal Society B: Biological Sciences, vol. 276(1670), pp. 3193-3200, 2009.

[17] A. Alexandra, P. Fanny, M. Allan, M. Ulrich and R. Michel, “Identification of visual paternity cues in humans”, Biology letters, vol. 10(4), 2014.

[18] L. T. Maloney, and M. F. Dal Martello F., “Kin recognition and the perceived facial similarity of children”, Journal of Vision, vol. 6(10), 2006.

[19] H. Yan, J. Lu, W. Deng, and X. Zhou, “Discriminative Multimetric Learning for Kinship Verification”, IEEE Transactions on Information Forensics and Security, vol. 9(7), 2014.

[20] R. Fang, K. D. Tang, N. Snavely, and T. Chen, “Towards computational models of kinship verification”, Proc. IEEE International Conference on Image Processing (ICIP), 2010, pp. 1577-1580.

[21] X. Qin, X. Tan, and S. Chen, “Tri-subjects kinship verification: Understanding the core of a family”, IAPR International Conference on Machine Vision Applications (MVA), pp. 580-583, 2015.

[22] J. Zhang, S. Xia, H, Pan, and A. K. Qin, “A genetics-motivated unsupervised model for tri-subject kinship verification”, IEEE International Conference on Image Processing (ICIP),pp. 2916-2920, 2016.

[23] J. P. Robinson, M. Shao, Y. Wu, H. Liu, T. Gillis and Y. Fu, “Visual Kinship Recognition of Families in the Wild”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40(11), pp. 2624–2637, 2018.

[24] R. F. Rachmadi, I. K. E. Purnama, S. M. S. Nugroho and Y. K. Suprapto, “Family-aware convolutional neural network for image-based kinship verification”, International Journal of Intelligent Engineering and Systems, vol 13(6), pp. 20–30, 2020.

[25] F. Schroff, D. Mathematical Problems in Engineering Kalenichenko and J. Philbin, “Facenet: A unified embedding for face recognition and clustering”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[26] L. Dulčić, “Face Recognition with FaceNet and MTCNN - Ars Futura”, https://arsfutura.com/magazine/face-recognition-with-facenet-and-mtcnn/, (Accessed on 21/10/2022).

[27] R. Fang, K. Tang, N. Snavely, and T Chen, “Towards computational models of kinship verification”, IEEE International Conference on Image Processing, pp. 1577-1580, 2010.

[28] G. Guo and X. Wang, “Kinship measurement on salient facial features”, IEEE Transactions on Instrumentation and Measurement, vol. 61(8), 2012.

[29] M. Xu and Y. Shang, “Kinship Verification Using Facial Images by Robust Similarity Learning”, Mathematical Problems in Engineering, pp. 1-8, 2016.

[30] “The Closed Eyes in the Wild (CEW) dataset”, http://parnec.nuaa.edu.cn/_upload/tpl/02/db/731/template731/pages/xtan/TSKinFace.html , (Accessed on 21/10/2022).

[31] J. Liang, Q. Hu, C. Dang, and W. Zuo, “Weighted graph embedding-based metric learning for kinship verification”, IEEE Transactions on Image Processing, vol. 28(3) pp. 1149–1162, 2019.

[32] R. Fang, A. C. Gallagher, T. Chen, and A. Loui, “Kinship classification by modeling facial feature heredity”, IEEE International Conference on Image Processing, pp. 2983-2987, 2013.

[33] T. H. Nguyen, H. H. Nguyen and H. Dao, “Recognizing families through images with pretrained encoder”, arXiv, 2020.

[34] Seldon, “Transfer learning for machine learning”, https://www.seldon.io/transfer-learning/, June 2021, (Accessed on 21/10/2022).

[35] S. Ronaghan, “The Mathematics of Decision Trees, Random Forest and Feature Importance in Scikit-learn and Spark”, Towards Data Science, https://towardsdatascience.com/the-mathematics-of-decision-trees-random-forest-and-feature-importance-in-scikit-learn-and-spark-f2861df67e3, (Accessed on 21/10/2022).

[36] “Advantages of a Decision Tree for Classification - Python”, https://pythonprogramminglanguage.com/what-are-the-advantages-of-using-a-decision-tree-for-classification/, (Accessed on 21/10/2022).

[37] S. M. Piryonesi and T. E. El-Diraby, “Role of Data Analytics in Infrastructure Asset Management: Overcoming Data Size and Quality Problems”, Journal of Transportation Engineering, Part B: Pavements, vol. 146(2), 2020.

[38] H. Ishwaran, “The effect of splitting on random forests”, Springer, vol. 99(1), pp. 75-118, 2015.

[39] S. Nembrini, I. R. König, M. Wright, “The revival of the Gini importance?”, Bioinformatics, vol. 34(21), pp. 3711-3718, 2018.

[40] N. Liberman, “Decision Trees and Random Forests”, Towards Data Science, https://towardsdatascience.com/decision-trees-and-random-forests-df0c3123f991, (Accessed on 21/10/2022).

[41] “4.2. Permutation feature importance — scikit-learn 1.0.1 documentation”, https://scikit-learn.org/stable/modules/permutation_importance.html, (Accessed on 21/10/2022).

[42] “Naive Bayes Classifier”, Machine Learning Simplilearn, https://www.simplilearn.com/tutorials/machine-learning-tutorial/naive-bayes-classifier, (Accessed on 21/10/2022).

[43] “Naive Bayes Classifier: Pros & Cons, Applications & Types Explained”, upGrad blog, https://www.upgrad.com/blog/naive-bayes-classifier/, (Accessed on 21/10/2022).

[44] A. Bakharia, “Visualising Top Features in Linear SVM with Scikit Learn and Matplotlib”, Medium, https://aneesha.medium.com/visualising-top-features-in-linear-svm-with-scikit-learn-and-matplotlib-3454ab18a14d, (Accessed on 21/10/2022)

[45] “SVM: Advantages Disadvantages and Applications”, Statinfer, https://statinfer.com/204-6-8-svm-advantages-disadvantages-applications/, (Accessed on 21/10/2022).

[46] “Advantages of Support Vector Machines (SVM)”, https://iq.opengenus.org/advantages-of-svm/, (Accessed on 21/10/2022).

[47] J. Cervantes, X. Li, W. Yu, and K. Li, “Support vector machine classification for large data sets via minimum enclosing ball clustering”, Neurocomputing, vol. 71(4), pp. 611-619, 2008.

[48] scikit-learn 1.0 documentation, https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html, (Accessed on 21/10/2022)

[49] “Advantages and Disadvantages of Logistic Regression”, https://iq.opengenus.org/advantages-and-disadvantages-of-logistic-regression/, (Accessed on 21/10/2022).

[50] “Advantages and Disadvantages of Logistic Regression - GeeksforGeeks”, https://www.geeksforgeeks.org/advantages-and-disadvantages-of-logistic-regression/, (Accessed on 21/10/2022).