Marks Fusion: Development Of Facial Marks Detection System And Fusion With Face Recognition System

Lucian Chirca
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
l.chirca@student.utwente.nl

1. ABSTRACT

Facial marks like freckles, moles, scars, pockmarks have been used in the past to identify individuals. There have been developed systems integrating both Facial Marks detection with Facial Recognition [17] [2], which showed improved performance over only using Facial Recognition. These systems used classic blob detection approaches like LoG (Laplacian of Gaussian) or Fast Radial Symmetry Transform for detecting facial marks, which gave a lot of False Positives, or had people manually annotate facial marks, which is too time consuming. Although there have been significant improvements in detecting Facial Marks using a Convolutional Neural Network, a system integrating this new approach with facial detection has not been implemented yet. This paper improves the state-of-the art in Facial Marks detection by using CNNs with deeper architectures and shows that a system combining a state-of-the-art algorithm in Facial Recognition with a Facial Marks Systems outperforms one that only uses Facial Recognition, especially in the case of monozygotic twins.

Keywords

Facial marks, Facial recognition, Convolutional Neural Networks, Monozygotic Twins

2. INTRODUCTION

Facial marks (e.g freckles, moles, scars, pockmarks, etc) are soft biometric features that have been shown to decrease the error rates in facial recognition software [8]. Although they are not discriminative enough by themselves to identify an individual, they have been proven to be effective at narrowing the search for the person [11], and helping to distinguish between people, especially in the case of monozygotic twins.[14]. Where Facial Recognition systems could output high scores, two people can have radically different facial marks patterns, not taken into account by a Facial Recognition system. This is especially true for identical twins, where it is very difficult to differentiate them only by using their facial structure, since they look so much alike. This is where facial marks can offer enough information to indicate if two pictures are from the same person or not. To be able to use facial marks effectively, their correct detection is crucial. Work has been done to show that a Shallow Convolutional Neural Networks (CNN) outperforms the classic blob detector approach such as Laplacian of Gaussian (LoG) or Fast Radial Symmetry Transform [15]. The topic facial recognition has gotten much attention, with a lot of research being done into improving the performance of such systems [9] [5] [12] [10]. The relevance of this topic is not surprising since facial recognition has many important implications such as being used for identifying suspects in relation to a crime, or being used as a way to identify an individual for security purposes. In this paper, we aim to improve the performance of a state-of-the-art face recognition algorithm, Open-face [1], by combining it with our facial marks system (FMS), and seeing if it can help differentiate between images of people, especially in the case of monozygotic twins. To this end, a subset of the FRGCv2 and 2009 and 2010 TwinsDays festivals in Twinsburg, Ohio datasets will be used and the following research questions will be addressed:

RQ1 To what extent does increasing the number of layers improve the performance of a facial marks detection CNN?

RQ2 To what extent is transfer learning better than creating CNN's from scratch.

RQ3 Can we, and to which extend, improve face recognition performance by fusing results obtained by the FMS, particularly in the case of monozygotic twins.

Experiment 1 will address Research Question 1 by comparing the performance of 4 CNNs with a shallow architecture (up to 3 layers) with 4 CNNs with a deeper architecture (between 3 and 9 layers). Experiment 2 will address Research Question 2 by comparing the performance of the 8 CNNs trained in Experiment 1 with a pre-trained deep CNN used for image recognition that has has the last 4 layers replaced and retrained for detecting facial marks. Lastly, Experiment 3 will use a CNN from the previous experiments to detect facial marks and fuse these scores with scores from a state-of-the art facial recognition software to try to get better results than just using facial recognition alone, especially in the case of monozygotic twins.

3. RELATED WORK

Facial marks have been used as a means of identification for a long time, the Bertillonage system being the first modern system to use facial features for identifying sus-

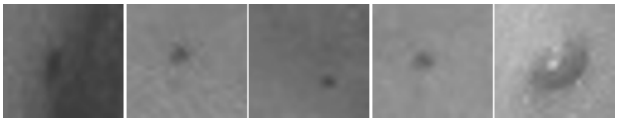


Figure 1. Example of facial mark patches

34th Twente Student Conference on IT Jan. 29th, 2021, Enschede, The Netherlands.

pects. [3] In Park and Jain [11], a facial marks detection system is implemented using classic blob detectors like Active Appearance Model (AAM) and Laplacian of Gaussian (LoG). This type of approach introduced a lot of false positives that needed to be filtered. In the case of [14], the case of identifying monozygotic twins is explored using classic blob detector methods like Fast Radial Symmetry transform. Since then, better Facial Marks detection systems have been made using Shallow Convolutional Neural Networks [15]. Grid-based approaches have been seen in [16] showing improved performance over classical methods. This novel approach surpassed the performance of traditional approaches such as blob detection with heuristics and had significantly less false positives. There are still ways to improve the performance of CNNs. As has been discussed in [13], creating deeper models is usually better than creating wider models. A model can have the same effective receptive field, with more layers, as has been shown [13]. Work has been done to show that a FMS can improve the performance of state-of-the-art facial recognition systems in [8] [2] [17]. Looking at the work that has been done so far, there seems to be a gap in using a CNN-based facial marks detection system with a state-of-the-art facial recognition system, such as [1], especially in the case of monozygotic twins, therefore this paper will focus on fusing these systems together and solving the difficult problem of identifying identical twins.

4. METHODOLOGY

4.1 Dataset

In this paper, from the FRGCv2 dataset, we will use the subset containing 12306 images of 568 subjects, in which the facial marks were manually annotated by [16]. The people in these images show a natural facial expression and were photographed under controlled conditions. This way, we provide our system with a relatively consistent dataset. The reason why we use this dataset is because it is sufficiently large, with a relatively consistent environment where the images were taken and has been manually annotated before. Otherwise, it will take too much time to manually annotate it ourselves. Additionally, we will use identical twin images acquired at the 2009 and 2010 Twins Days Festival in Twinsburg, Ohio, to test our system in the case of monozygotic twins identification. From this dataset, 100 pairs of identical twins will be selected, resulting in 200 people and for each person 2 pictures will be extracted. This will result on 400 images to be used in experiments. These images will be processed the same as the first dataset. The reason 2 images are extracted for each person is so that we have enough images to see if the system can recognize if two images are from the same person or from different people.

4.2 Image pre-processing

For the image pre-processing part, we apply geometric and photometric transformations to the images prior to the facial marks detection and facial recognition steps. We crop the images to (800,600) and use the affine transformation that will map the pupils to the coordinates (200,250) and (400,250), such that the inter-pupillary distance for every image is the same (200 px). [15] This gives images a consistent coordinate system in which to store the locations of detected facial marks. It also provides consistency between images of a person taken in different environments. For the photometric transformation, we apply a grayscale transformation to reduce the complexity of the problem. After this we normalize the image by subtracting a mean of 0.5 and dividing with a standard deviation of 0.5. This is done to reduce the time to converge. Figure 2 shows an example of this procedure.

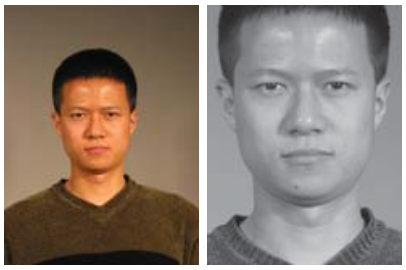


Figure 2. Image 1 - Original, Image 2 - After transformation

4.3 Patch generation

After pre-processing, the dataset is split into training and evaluation sets. The images from the first 390 people will be used for training and the rest will be used for evaluation. For each of these sets, we will extract 10000 skin patches containing facial marks and 50000 skin patches not containing facial marks. This procedure will be repeated for 3 different skin patch sizes: 15×15 px, 19×19 px and 25×25 px. These collections will be used for training and evaluating models in Experiment 1 and Experiment 2. Skin patches containing facial marks will be extracted according to the locations that have already been manually annotated. To generate skin patches not containing facial marks, firstly the face of the subject will be detected and withing that bounding box skin patches will be randomly selected such that a) they don't overlap with facial marks and b) they don't overlap with already selected patches.

4.4 Facial mark detection

To detect the facial mark pattern on a person's face, we shall use a grid-based approach that will divide the face into a grid of equal-sized rectangles. For each rectangle, we will use our CNN-based classifier to detect the facial mark. Once it is detected, the result will be added to the set containing the facial marks of the person's face, along with the location of the facial mark. [16] The size of each rectangle will be decided based on experimental results, but we expect that it should be large enough to contain a relatively large facial mark and small enough to separate small facial marks close to each other.

4.5 Facial marks matching

Once facial marks have been detected, it is important to calculate how different two images are given the facial mark locations. Given an image, we split it into a grid of *rows* x *columns*, where the size of each rectangle is determined by dimensions of the face bounding box split over the number of rows and columns. The best grid configuration will be found during Experiment 3, based on empirical results, from 4 different categories: coarsest, coarse, finer and finest, defined by setting the number of rows and columns to the aspect ratio of the image (4×3), multiplied by 2,5,8 and 11, respectively. Then we run our CNN model in a "sliding window" fashion, whereby we classify skin patches taken distance(stride) d from adjacent patches, such that some visual data may overlap. If the center of a skin patch classified as having a facial mark is within a rectangle in a grid, then that rectangle gets the score 1, indicating that the rectangle contains at least one facial mark, otherwise the score is 0. After this grid has been established for two images, we can compute the negative hamming distance between them. This distance will then be used to determine how similar the facial

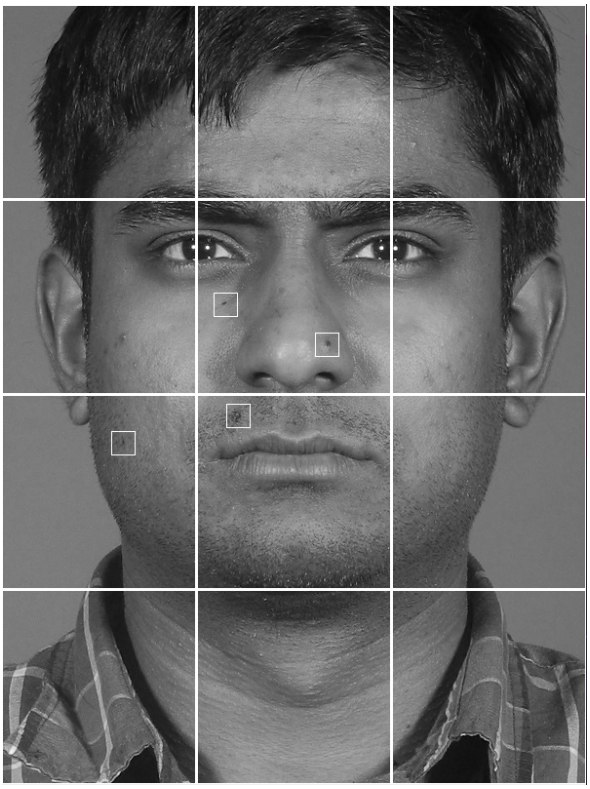


Figure 3. Example of a facial matching 3x4 grid

mark pattern between two images is. An example of such a grid can be seen in Figure 3.

4.6 Fusing algorithm

To combine the similarity scores between the Facial Recognition (FR) system and the Facial Marks System (FMS), it is important that we first normalize the outputs of each, in such a way that they are in the same range.

To normalize the Facial Mark System score, we will perform min-max normalization on the negative hamming distance between the facial grids of two images. This will result in a score from 0 to 1, whether the higher the score, the more likely the two images have the same facial mark pattern.

For the Facial Recognition system, we will obtain the feature vector containing 2D and 3D facial features of two images, v_1 and v_2, respectively. We will perform the following operation to get the angle between the two vectors: $angle = \frac{\arccos(v_1 \cdot v_2)}{|v_1||v_2|}$. If we divide this angle by π and subtract 1 by this value, we get a score from 0 to 1 compatible with the FMS score.

After this, we compute a final similarity score as follows: $SC = w_1 * FR + w_2 * FMS$, where $w_1 + w_2 = 1$. Experimental results should show what are the best weights for the combined system.

4.7 Experimental setup

In this paper, we use Receiver Operating Characteristic (ROC) curves and Equal Error Rate (EER) to measure performance and compare with previous results. The EER is the point on the ROC curve where the false acceptance rate and the false rejection rate are equal. Lower EER indicates better performance of the classifier. The ROC curve shall show the True Positive and False Positive rates for all classification thresholds. The above metrics will be used to evaluate each model.

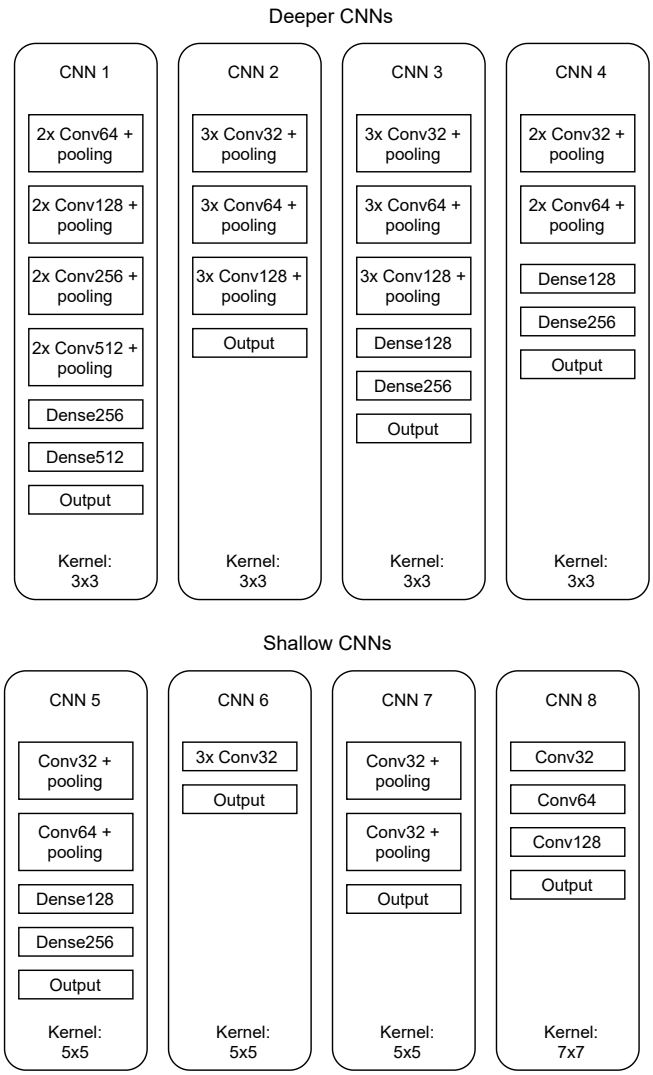


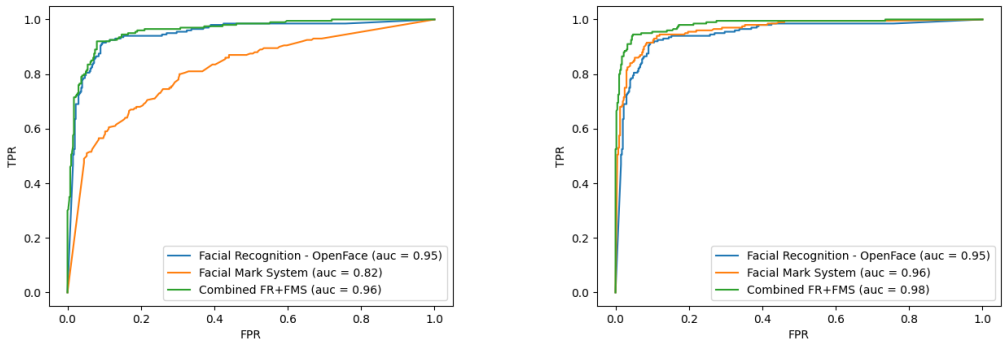
Figure 4. CNN 1-8

At the end of each CNN, the output of the last layer will be passed through a Softmax activation function to compute the probability that a patch contains a facial mark. This output will be passed through a threshold. If the probability that a patch contains a facial mark is higher than the threshold then the output is $True$, otherwise the model outputs $False$.

Experiment 1 - Increasing the number of layers

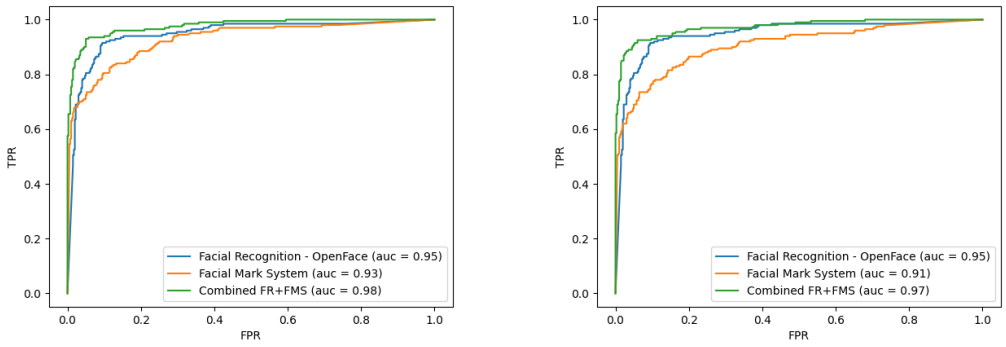
The aim of this experiment is to compare the performance in detecting facial marks of Convolutional Neural Networks which have up to 3 layers (shallow) with CNNs which have between 3 and 9 layers (deeper). For this purpose, 8 CNNs will be developed (4 shallow and 4 deeper). Each model will be trained from scratch for 4 epochs and evaluated on 3 different skin patch sizes: 15×15 px, 19×19 px and 25×25 px. The kernel size for each model will range between 3 and 7. The performance of each kernel size will also be evaluated. Padding is added to each convolutional layer if the input image is too small. The stride is fixed for all models at 1.

The deeper models are inspired by the VGG [13] architecture. They use stacked 3×3 convolutional filters instead of larger filters like 5×5 or 7×7, followed by a pooling layer, since stacking two or more 3×3 filters has the same effective receptive field as a 5×5 or larger filter, respectively,



a) Coarsest - 8×6 grid

b) Less coarse - 20×15 grid



c) Finer - 32×24 grid

d) Finest - 44×33

Figure 5. ROC curves for Experiment 3 performed on the FRGCv2 dataset.

with the added performance of a deeper model.

Furthermore, the convolutional layers increase in width by a factor of 2 each time, since earlier layers don't have that many features to learn. The first layers only need to detect simple features like lines or blobs.

In Figure 8, we can see a selection of trained 7×7 filters in the first convolutional layers. These filters clearly shows that the model has learned to detect blobs, which is what facial marks tend to look like.

In some of the deeper network (CNN 1,2 and 3), batch normalization [7] will be used to accelerate model training and prevent the vanishing gradient problem, whereby because of the depth of the network, the gradient becomes too small in some layers and the weights don't update properly. The best model resulting from experiments will be used further to generate a facial mark pattern of a person's face.

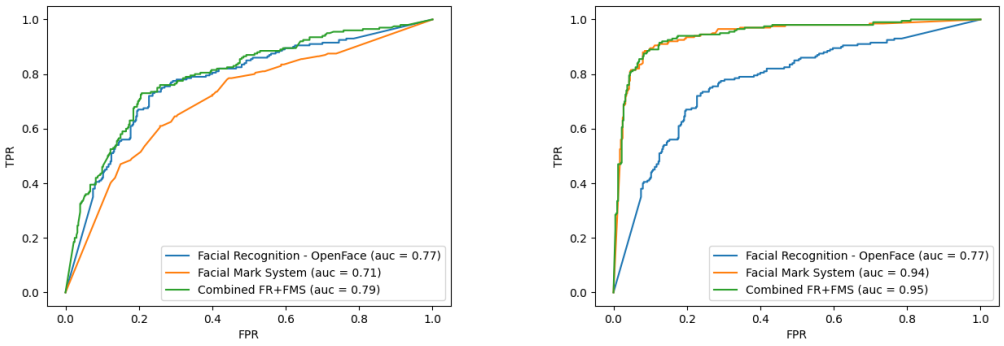
Experiment 2 - Comparing transfer learning with training from scratch.

Transfer learning will be explored in this experiment by loading a model of the MobileNet [6] architecture that has been pre-trained on the ImageNet dataset [4], freezing all convolutional layers and replacing the last 4 fully connected layers with fully connected layers of sizes 1024, 1024, 512, 2, respectively, where the 2 at the end represents the 2 classes that we wish to predict: patch contains facial mark or doesn't. These last 4 fully connected layers shall be trained for 4 epochs and then the performance will be compared with the other 8 CNN architectures from the previous experiment. The idea behind this is that the pre-trained MobileNet model likely already captures image features which we wish to extract (lines, edges, blobs, etc.). By only training the fully connected layers at the end and reusing the filters, we teach the model to use those filters for a new task, thus decreasing the time it takes to train a deep CNN such as MobileNet, which has 28 layers. Since MobileNet accepts input images of size 224×224 px with 3 color channels, but we have skin patches of sizes 15×15, 19×19 and 25×25 pixels with one color channel, we would have to upscale images to the correct input size and changing the number of color channels from 1 to 3, by copying over the grayscale channel and obtaining an RGB image. This obviously means that the network won't be able to leverage the colors of images, but it is still expected that the network would be able to extract valuable features from the images.

Experiment 3 - Fusion

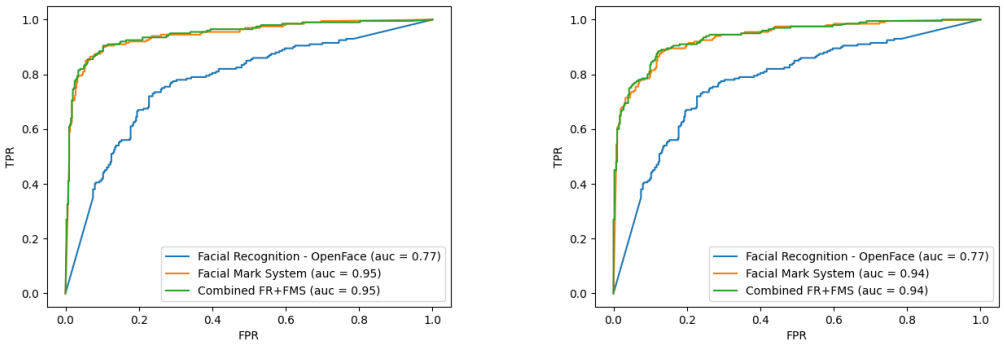
In this experiment, we will fuse the Facial Mark System (FMS) with the Facial Recognition (FR) system. Each of these systems will be tried in the following scenarios: FR alone, FMS alone and the fusion of FMS and FR. The results will then be compared. Furthermore, this experiment will be performed on both datasets for comparison. For fusing the two systems together, the weights of each system will be selected such that the Equal Error Rate of the resulting predictions is minimal.

For the Facial Mark System, 4 different grid configurations will be tried. Also, for the FMS, a stride of 5 pixels will be used for detecting marks. This means facial patches will be extracted 5 pixels apart from adjacent patches, in order to maximize the number of facial marks detected. There will be cases when a single facial mark will be detected multiple times, but that will be mitigated by the superimposed grid which will take multiple facial mark patches and aggregate them into a score of 1 or 0. To reduce the number of false positives (from clothing, jewelry, hair, etc.) like in Figure 12, we superimposed a rectangle where



a) Coarsest - 8×6 grid

b) Less coarse - 20×15 grid



c) Finer - 32×24 grid

d) Finest - 44×33

Figure 6. ROC curves for Experiment 3 performed on the Ohio Twins Day festival dataset.

approximately the face will be and where the facial marks will be extracted from. Since all faces were centered and the eyes were mapped in the same location, this can be done.

For each pair of twins, twin A and twin B, there is a total of 4 images, A_1, A_2, B_1, B_2, respectively. Each of the above system will perform all combinations of two elements from the 4 images (6 in total). This will result in two cases where two images of the same person is compared and four cases where images of different people are compared. Since we take 100 pairs of twins, this will result in 600 scores. The same procedure will be performed on the FRGCv2 dataset where we will select 100 pairs of non-twins, resulting in 600 scores. These scores will be used to compare the 3 scenarios described above. From this, it can be seen if fusing the FMS with the FR system actually gives better results.

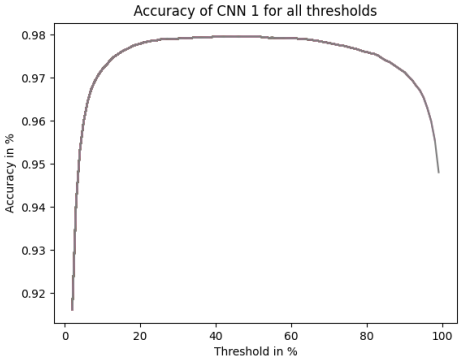


Figure 7. Accuracy of CNN 1 for all thresholds.

5. RESULTS AND DISCUSSION

5.1 Experiment 1 - Deeper architecture

Table 1 shows the performance of the 8 CNN architectures ran on 3 different patch sizes. The lowest EER for a patch size is underlined. It can be seen that for each model, the error is higher with each decreasing patch size, thus indicating that decreasing the patch size has a negative influence on the performance. Furthermore, for all patch sizes, it can be seen that deeper models outperform shallow models. For all patch sizes, the best performance is offered by the deeper models.

For the threshold, based on empirical results, a value 50% was selected. Since the output of the system is a probability between 0 and 1, a 50% threshold means that if the output of the neuron corresponding to output "$True$" is higher than the 50%, then the resulting prediction will be $True$. This result is achieved by looking at the accuracy of CNN 1 in Figure 7 for all threshold values from 2% to 99%. The reason for these bounds is to exclude the outliers so the figure is clearer. At 50%, the system has the highest accuracy overall.

If we took the point where the False Negative Rate is equal to the False Positive Rate, we would have a value of 71% for the threshold. This would give us a True Positive and True Negative rate of 96%, which is undesirable, since skin patches containing facial marks are far fewer than patches not containing facial marks, thus we would have a lot of False Positives, which will lower the performance of the system.

From the shallow models, it seems that the kernel size of 7×7 has similar EER to best performing shallow models

evaluated on patches of sizes 15×15 and 25×25 and the lowest EER, evaluated on patches of size 19×19 px. This indicates that a filter size of 7×7 is preferred over a 5×5 filter. Despite this, the deeper models incorporating 3×3 filter sizes still outperform the CNNs with large filter sizes. This is explained by the fact that stacking layers with 3×3 convolution filters achieves the same effective receptive field with better performance than single layers with large filters. [13]

CNN 4 and CNN 5 have similar architectures except for the face that the convolutional layers in CNN 5 with filters of size 5×5 are replaced in CNN 4 by stacked layers with 3×3 filters, as explained in the VGG paper [13]. It can be seen that indeed the deeper version has a very close or lower error for patches all patch sizes. The same argument goes for CNN 2 and CNN 8, which have also similar architectures and uphold the above idea.

It can thus be concluded, from the results of this experiment, that adding more convolutional layers to a Convolutional Neural Network increases its performance for the task of detecting facial marks. The improvement is not extreme, but it does improve upon the current state of the art in facial mark detection. One explanation why the improvement is not as dramatic as the improvement from the classical way of using blob detectors to using CNNs is that it is a relatively simple task to detect a facial mark and it does not require a very complicated model.

5.2 Experiment 2 - Transfer learning

In this experiment, transfer learning was performed on the MobileNet model pre-trained on the ImageNet dataset by replacing the old fully connected layers at the end with 4 dense layers of sizes 1024, 1024, 512 and 2, respectively, and training those layers from scratch. This was done on skin patches of size 25×25 px, 10000 of which contained facial marks and 50000 of which didn't. The reason other patch sizes were not included in the experiment is because it has already been established in the previous experiment that decreasing the patch size monotonically decreases the performance.

The evaluation was done with a collection of the same number of skin patches but from different people. The resulting EER is **0.043**. Looking at Table 1, we can see that the error is lower than CNNs 6 and 7, but higher than the rest of the models. This means that greater performance for the task of facial mark detection can be achieved from models created in Experiment 1 than by doing transfer learning. This may have been caused that the fact that the model was pre-trained on color images, but we give it grayscale images, thus reducing its performance, since the model cannot use the color information it has been trained with. Additionally, although images are upscaled from 25×25 px to 224×224 px, those images still have the same amount of data in them as the smaller counterpart, while the model expects images of size 224×224 that have not been upscaled. This means that we are doing something the model has not been trained on, thus the loss in performance. However, the performance is still acceptable, since the error is not significantly higher than the other models. This is likely due to the fact that the pre-trained model still is able to extract useful features like lines, blobs, edges to detect facial marks. This is the reason for re-training the fully connected layers at the end: to use these learned features to learn what a facial mark looks like.

5.3 Experiment 3 - Fusion

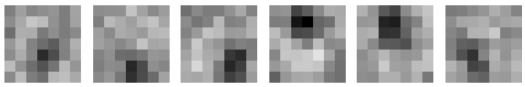


Figure 8. Selection of filters from CNN 8

Table 1. EER of all classifiers.

Classifier	15×15	19×19	25×25
CNN 1	0.0505	0.0449	0.0315
CNN 2	0.0556	0.0397	0.0376
CNN 3	0.0654	0.0392	0.0325
CNN 4	0.0602	0.0589	0.0346
CNN 5	0.0634	0.0548	0.0382
CNN 6	0.0712	0.0547	0.0470
CNN 7	0.0684	0.0656	0.0474
CNN 8	0.0648	0.0466	0.0386

As can be seen in the ROC curves in Figure 5 and Figure 6, OpenFace has a higher Area Under the Curve (AUC) for the FRGCv2 dataset, than on the Twins dataset. Its EER on the non-twins dataset is 0.0937 compared to 0.2625, on the twins dataset. This is approximately a 64.3% relative difference. The reasoning for this is that, since we are comparing monozygotic twins, the Facial Recognition system will output very similar scores while comparing images of identical twins. This is explained by the fact that twins have very similar facial properties, which is what makes them a difficult case for Facial Recognition.

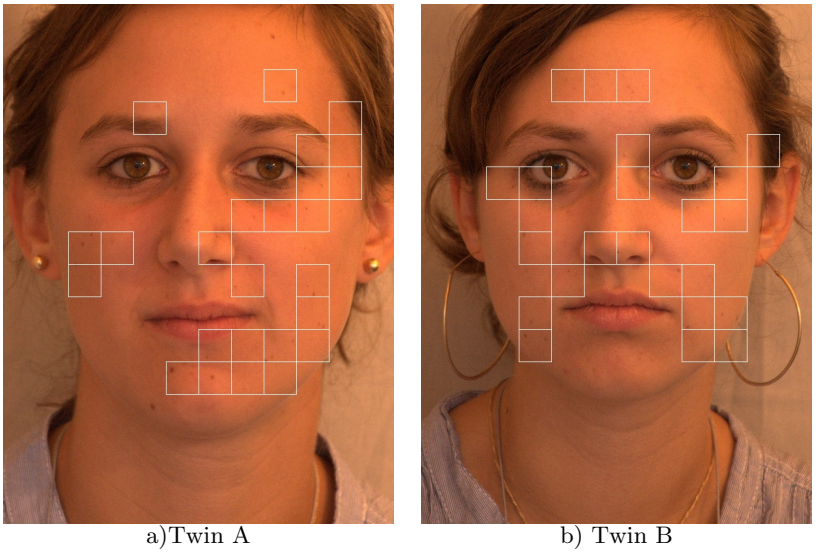
We also observe, that there is a difference in the performance of the FMS between the two datasets, for the same grid configurations, but this difference is not as dramatic (at most 21.3% relative difference for grid configuration 8×6) as the difference described above. This could be explained by the fact that, as was discussed in [14], the facial mark patters between twins do appear to be correlated.

In Figure 9, we can see a pair of twins with different grid configurations generated by the FMS. It can be seen that the two grids are similar, but have important differences, especially around the chin area, where Twin A clearly has a few moles, while Twin B doesn't. This information can help distinguish between twins, and the rectangles that are the same for both twins don't affect the negative hamming distance. This indicates that there differences between the facial mark patterns of twins that can be used to improve facial recognition.

Seeing the ROC curves in Figure 6, we note that the Facial Mark System has a significantly higher performance than the Facial Recognition system (OpenFace) run on the twins dataset,(up to 61.4% relative difference for grid configuration 32×24) for all grid configurations except the coarsest. This exception is due to the fact that a 8×6 grid does not capture enough facial mark informa-

Table 2. EER of Experiment 3 done on the FRGCv2 dataset for different grid configurations.

Grid	FMS weight	EER of FMS+FR	EER of FMS
8×6	0.2	0.08	0.2575
20×15	0.37	0.0512	0.085
32×24	0.41	0.0612	0.16
44×33	0.8	0.0675	0.1687



a)Twin A b) Twin B

Figure 9. Differences in facial mark grids between twins for grid size 12×9.

Table 3. EER of Experiment 3 done on the Twins dataset for different grid configurations.

Grid	FMS weight	EER of FMS+FR	EER of FMS
8×6	0.16	0.25	0.3275
10×8	0.62	0.235	0.2875
20×15	0.9	0.1041	0.1062
32×24	0.9	0.0987	0.1012
44×33	0.8	0.12	0.125

tion to be useful. This fact can also be observed in Figure 5, where this grid configuration under performs, due to it being too coarse. It can thus be concluded, that the Facial Mark System significantly outperforms the Facial Recognition system for differentiating between monozygotic twins. However, looking at Table 2 we see that for the best performing grid configuration, the Facial Marks System has approximately the same performance as the Facial Recognition system.

Looking at the combination of the Facial Mark System with the Facial Recognition system, in Figure 5 and in Table 2 we observe that the combined system does offer the best performance in some cases, with a 39.7% relative decrease in error rate for grid dimensions 20×15, from 8.5%, given by the Facial Mark System alone, to 5.12%. For the coarsest grid configuration, there is no significant performance improvement for the combined system. This is likely due to the fact that this grid size does not provide enough useful information to help the classification, since its error rate is so high compared to OpenFace.

When this experiment is run on the Twins dataset, however, we see in Figure 6, no grid configuration offers a significant improvement for the combined system. For the coarsest grid, we note a 4.7% relative reduction in error from 26.5% achieved by OpenFace, to 25%, which is rather insignificant compared to the results for the FRGCv2 dataset. In Figure 10 and Table 3, for the grid configuration 10×8, the combined system run on the Twins dataset has the lowest error rate, with a relative reduction in EER of 10.4% from 26.25%, obtained by OpenFace, to 23.5%.

We observe that, for the Twins dataset, the combined system offers an increase in performance when the difference

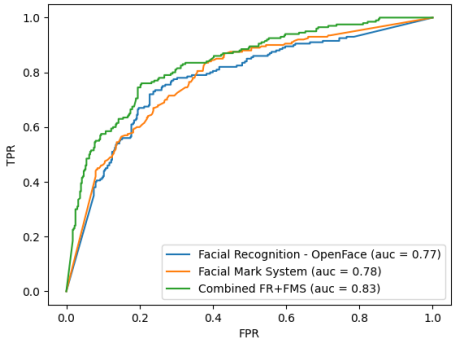


Figure 10. ROC curve on the twins dataset for grid configuration 10x8.

in error rates between the Facial Mark System and Facial Recognition system are not significant, such as for grid configurations 8×6 and 10×8, with relative differences of 20% and 12.2%, respectively. This fact is also observed for the combined system run on the FRGCv2 dataset, with all grid configurations showing a better performance for the combination of the systems, except for the coarsest grid size, where the relative difference in error rate between the FMS and OpenFace is 63.3%, which is significant.

This is because when one system significantly outperforms the other in terms of error rates, the under-performing system's output is much closer to random noise than to useful information and thus drags down the performance of the combined output. When the Facial Mark System and Facial Recognition systems have similar performance, they both provide different useful information about the face, thus resulting in a prediction that is better than the individual systems' predictions.

In Tables 2 and 3, the weights are shown for each experiment. They were selected such that the EER of the combined systems is minimal, on the corresponding dataset, for the specific grid configuration. In the case of the experiment performed on the Twins dataset, we clearly see in Table 3,that the weights are generally higher, in some case even more than double than those in Table 2. This is an indication that the combined system (FMS+FR) re-

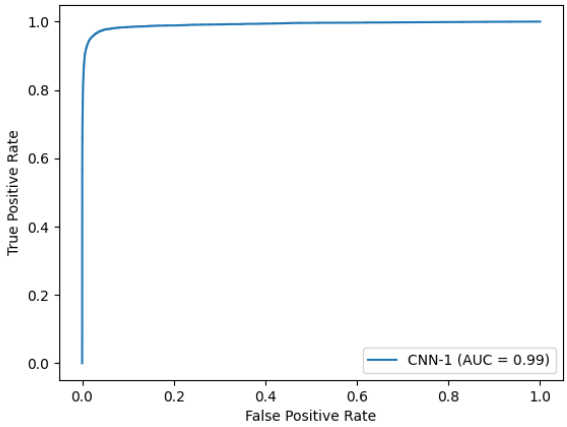


Figure 11. ROC curve for the best performing CNN in Experiment 1.

lies more on the predictions of the Facial Mark System in the case of the twins dataset than in the case of the FRGCv2. This is explained by the fact that facial marks have a stronger discriminating power on the twins dataset than on the non-twins dataset, due to how poorly the Facial Recognition system differentiates between twins, since they have very similar facial features.

As for the grid configurations, we can clearly see that the performance is lowest for both Tables 2 and 3 in the case of the coarsest grid, reaches a peak for grid configuration 32×24 for Table 3 and 20×15 for Table 2, then decreases for the finest grid configuration. This is explained by the fact that a very coarse grid groups too many facial marks in a large rectangle to offer much useful information about the facial mark pattern. Furthermore, a very fine grid is prone to errors due to small differences in alignment, position or size of the face. The best performance is offered by a grid configuration in between these extremes.

From this experiment we can conclude that for identifying if two images of non-twins belong to the same person, a system combining a Facial Mark System and a state-of-the art Facial Recognition software like OpenFace, offers the best performance. However, for images of twins, the best performance is offered by a Facial Mark System, since Facial Recognition severely under-performs in this scenario.

5.4 Conclusions and Further work

In this paper, we were able to improve the performance of the state of the art Facial Marks System by adding more layers to the Convolutional Neural Networks used for facial mark classification. We noted that the results weren't as dramatic as the improvement from the classical method of using blob detections such as LoG or Fast Radial Symmetry Transform to using Convolutional Neural Networks for facial mark detection. Nonetheless, Experiment 1 showed that there is a noticeable improvement in detecting facial marks with deeper networks than more shallow networks.

For the future, the CNNs can be improved by adding layers with 1x1 kernel size, which would act as linear transformations of the previous layer and have been shown in [13] to increase performance in some deep CNNs.

For Experiment 2, we observed that transfer learning in our case did not show a decrease in error rate compared to Experiment 1, but the results were not significantly worse, either.

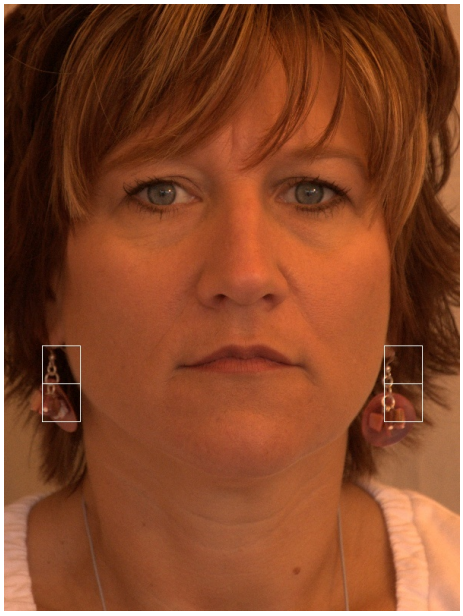


Figure 12. Example of False Positives.

Lastly, for Experiment 3, we observed that the Facial Marks System developed in Experiment 1 had significantly better results than a state-of-the art Facial Recognition software, OpenFace, for the task of differentiating between monozygotic twins. In the case of the FRGCv2 dataset, fusing the FMS with the FR system showed better results than just using the Facial Marks System. We did observe that fusing the two systems did not offer a performance improvement when run on the Twins dataset.

To mitigate this problem, one may try to combine the two systems in a different way. For example, when the scores of the Facial Recognition system are very close to each other, or when it is known that the system will be used to differentiate twins, the weights, instead of being manually entered, could be automatically adjusted to favor the FMS scores, and otherwise shift the weights towards the Facial Recognition scores.

For the future, to reduce the problem of false positives like in Figure 12, one could add to the training set patches containing images of things that usually cause false positives, such as jewelry or clothing.

6. REFERENCES

[1] Baltrusaitis, T., Robinson, P., and Morency, L.-P. Openface: An open source facial behavior analysis toolkit. pp. 1–10.

[2] Becerra-Riera, F., Morales-González, A., and Vazquez, H. Facial marks for improving face recognition. *Pattern Recognition Letters 113* (05 2017).

[3] Bertillon, A. Identification anthropometrique: Instructions signaletiques., 1893.

[4] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09* (2009).

[5] Deng, J., Guo, J., Xue, N., and Zafeiriou, S. Arcface: Additive angular margin loss for deep face recognition, 2019.

[6] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. Mobilenets:

Efficient convolutional neural networks for mobile vision applications, 2017.

[7] Ioffe, S., and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015.

[8] Jain, A. K., and Park, U. Facial marks: Soft biometric for face recognition. In *2009 16th IEEE International Conference on Image Processing (ICIP)* (2009), pp. 37–40.

[9] Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems* (2012), F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., vol. 25, Curran Associates, Inc., pp. 1097–1105.

[10] Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., and Song, L. Sphereface: Deep hypersphere embedding for face recognition, 2018.

[11] Park, U., and Jain, A. K. Face matching and retrieval using soft biometrics. *IEEE Transactions on Information Forensics and Security 5*, 3 (2010), 406–415.

[12] Schroff, F., Kalenichenko, D., and Philbin, J. Facenet: A unified embedding for face recognition and clustering. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Jun 2015).

[13] Simonyan, K., and Zisserman, A. Very deep convolutional networks for large-scale image recognition, 2015.

[14] Srinivas, N., Aggarwal, G., Flynn, P. J., and Vorder Bruegge, R. W. Analysis of facial marks to distinguish between identical twins. *IEEE Transactions on Information Forensics and Security 7*, 5 (2012), 1536–1550.

[15] Zeinstra, C., and Haasnoot, E. Shallow cnns for the reliable detection of facial marks. In *2018 International Conference of the Biometrics Special Interest Group (BIOSIG)* (2018), pp. 1–5.

[16] Zeinstra, C., Veldhuis, R., and Spreeuwers, L. Grid-based likelihood ratio classifiers for the comparison of facial marks. *IEEE Transactions on Information Forensics and Security 13*, 1 (2018), 253–264.

[17] Zhang, Z., Tulyakov, S., and Govindaraju, V. Combining facial skin mark and eigenfaces for face recognition. In *Advances in Biometrics* (Berlin, Heidelberg, 2009), M. Tistarelli and M. S. Nixon, Eds., Springer Berlin Heidelberg, pp. 424–433.