Investigating topics in Facial Recognition

[An ARI3129 Assignment]

Charlton Sammut
Faculty of Information and Communication Technology
University of Malta
Msida, Malta
charlton.sammut.18@um.edu.mt

***Abstract*—The face is one of the primary ways humans identify each other. The ability for humans to not only recognise others by their face but also their current emotional state is quite an impressive feat, and yet it is done flawlessly and with little effort.**

Face recognition can be defined as the process of identifying an individual using only their face. This report investigates current face recognition by comparing four state-of-the-art facial recognition algorithms.

***Index Terms*—Facial Recognition, Artificial Intelligence, Artificial Vision, Deep Learning, Person Identification**

I. Introduction

Facial recognition, also refereed to as face recognition is one of the most popular topics in computer vision [1]. This was clearly seen in 2020 during the iEEE conference on computer vision and pattern recognition [2], where face, gesture, and body pose recognition ranked as the 3rd most popular research area [3]. Facial recognition has been applied to many real-world domains such as forensics [4], human computer interaction [5] and security [5].

Face recognition can be defined as a problem where, given an image i containing the face of an unknown person p and a set X of labeled images, where each image contains a face of a known individual. Assuming that X contains some number of images of person p, how can one determine the identity of the person a in image i [1].

II. Aims And Objectives

The main research question that this report aims to answer is ,"Given an image containing the face of a person p, how accurately can current state-of-the-art facial recognition algorithms identify that person p from a set of faces X?" This research question is going to be answered by completing the following 2 objectives:

- Objective 1 (O1): Compare and contrast different labeled data-sets that contain images of people's faces in order to find the most suitable available data-set.
- Object 2 (O2): Compare and contrast different state-of-the-art facial recognition algorithms in order to find the most suitable four. The chosen four facial recognition algorithms will then be implemented and applied to the data-set chosen in O1.

In O1 a number of different labeled data-sets are going to be compared and contrasted on a number of different criteria. In O2, two traditional facial recognition algorithms are going to be selected alongside two deep learning facial recognition algorithms. For the sake of fair testing, the chosen models will all be applied on the same data-set that will be chosen in O1.

III. Literature Review

In order to apply face recognition on an image i, the following steps need to be achieved: [6]

- Step 1: Apply face detection on image i.
- Step 2: Extract the faces detected in step 1.
- Step 3: Classify all of the extracted faces.

Face detection is the first step in the face recognition process. This step determines whether the image contains a face. In general, face detection is defined as the task of detecting and localizing an unknown number of faces from a given image [7].

If a face is detected, then it is extracted in the second step. It is important that the way the face is extracted follows an agreed upon structure [6]. An example of such a structure is the bounding box structure [8].
The final step is the face classification step. There are two types of face classification, the first kind is refereed to as face verification [6]. Face verification is a boolean problem, where one needs to check whether the extracted face is the same as one other known face. The second type is refereed to as face identification [6]. In the problem of face identification, one needs to identify the extracted face by comparing it to several other labeled faces. This report will mainly focus on the problem of face identification rather then face verification, however many of the techniques used to tackle face identification can also be used to tackle face verification.

Many different solutions have been proposed to tackle the problem of face recognition, however most of the proposed solutions can be clustered into one of the following four sub-classes: [6]

- Holistic Approaches
- Geometric Approaches

- Local-Texture Approaches
- Deep Learning Approaches

Holistic approaches assume that in an image containing a face, there are a number of redundant features that can be removed via tensor decomposition [6], [9]. When all the redundant features are removed and only the core features of the face are left, the distance between the core features and all the faces in the set of possible identifies is calculated. The identify of the image that has the shortest distance is returned. Popular techniques that fall under this category include:

- Eigenfaces, also known as Principal Component Analysis (PCA) [10].
- Fisherface, also known as Linear Discriminative Analysis (LDA) [11].
- Independent component analysis [12].

While very high accuracy scores have been achieved through holistic approaches they tend to be very susceptible to misalignments and context changes [6].

Geometric approaches use the distribution of crucial face landmarks to calculate certain facial data, such as the distance and angle of two given landmarks [6], [13]. In past literature, it has been observed that certain face landmarks are much better to use for facial recognition then others [6], [13]. For example, the cheeks and the forehead have very simple structures and fewer distinctive attributes when compared to more complex features like the eyes or the nose. Popular techniques that fall under this category include:

- Elastic graph matching [14].
- Dense local graph structure [15].

The biggest disadvantages that comes with using a geometric approach is that all the images need to be perfectly aligned. This implies that if the entered images are not aligned, computational power has to be used on aligning the images [6].

Local-Texture approaches are somewhat similar to geometric approaches, but rather then using the distribution of features to calculate facial data, they extract the textures of the individual face landmarks. Unlike geometric approaches this makes them very resilient to variance in lighting and brightness. Another benefit of using local-texture approaches over geometric is that unlike geometric approaches, local-textures don't require that the entered images are aligned [6]. Popular techniques that fall under this category include:

- Local binary pattern [16].
- Binarized statistical image features [17].

While the advantages of local-texture approaches are clear, they tend to suffer when the entered images have a lot of varying facial expression [6]. This is due to the fact that when making different facial expression, attributes of facial features tend to change.

Deep Learning approaches make use of structures called artificial neural networks. These are used to solve a given problem and tend to take inspiration from the human brain. Recent applications of deep learning on the problem of face recognition show promising results. Such implementations include:

- DeepFace [18].
- COCO Loss [19].
- ArcFace [20].

While the benefits of deep learning can clearly be seen, it also comes with its own set of downfalls. When compared to traditional techniques, deep learning approaches tend to be more computationally heavy and require much more data.

In order for fair comparisons of different face recognition models to be made, a data-set needs to be selected. A good face recognition data-set should have a good number of images with various different conditions, such as different facial expressions under different lighting conditions. One should also keep it mind the size of the data-set. The chosen data-set needs to be large enough to contain enough images so that the models can be properly trained and evaluated, while at the same time being small enough to keep training time to a reasonable duration. When choosing a data-set this was constantly considered as a total of four models need to be trained in order for evaluation to properly occur.

A review conducted in 2020 compared and contrasted 23 different data-sets that were created to help tackle the problem of face recognition [6]. From these 23 different data-sets the following stood out:

- The Olivetti Research Laboratory (ORL) data-set: 400 images of 40 different people with varying conditions of illumination, varying hairstyles and different facial expression [6], [21].
- The Labeled Faces in the Wild (LFW) data-set: 13,233 images from 5749 individuals. All images were collected from wild i.e. collected from totally uncontrolled conditions [6], [22].
- The Celebrities in Frontal-Profile (CFP) data-set: 7000 images of 500 different individuals [6], [23].

The other mentioned data-sets were discarded as they were either not publicly available or they contained to many images or their images did not have enough varying conditions.

IV. Methodology

In this section of the report, the theory behind each objective is first going to be explained, after which the implementation process is going to be explained. All of the implementations mentioned in this report were done using Python [24].

A. *Objective 1*

At first the LFW data-set was going to chosen, but upon closer inspection it was observed that the majority of the individuals in the data-set only have one image. Since to conduct a face recognition evaluation on an individual one needs at least two images, this data-set was also discarded. After great consideration the ORL data-set was selected. It was chosen due to its many benefits, such as having a variety of faces in terms of gender, age and ethnicity. All of the 40 individuals inside of the data-set can be seen with different facial expressions under different lighting conditions.

Fig. 1. Faces from the ORL data-set [21]

.

The ORL data-set is publicly available as a .csv file [25]. In order for the models to easily read the data-set, a python file named *generateImages* was created in order to transform the images from a .csv format to .jpg format.

B. Objective 2

The same previously mentioned review, compared and contrasted face recognition models from all of the four mentioned approaches [6]. In the review it was concluded that while all of the approaches have their advantages and disadvantages, holistic approaches tend to give good results even when applied on various different data-sets with varying conditions.

When it comes to holistic approaches, eigenfaces are considered to be a pioneering method [6]. This, combined with the fact that PCA has previously given good results on a number of different data-sets, make it a very good candidate for O2 [6], [10]. Hence it was chosen to be one of the models to be used for O2. Another popular technique is fisherface. In past literature fisherface has been extensively compared to eigenfaces [26], and results show that depending on the conditions of the data-set, fisherface could give better results [26]. This is especially true for data-sets that have variations in lighting conditions [26]. Due to this fact, fisherface was chosen as the second model to be used for O2.

Apart from holistic approaches, deep learning approaches also show a lot of promise. In recent literature, deep learning techniques have been applied to a variety of problems, with some applications achieving success rates that were never observed before [27]. In the same review paper [6], 37 different deep learning algorithms that were previously applied to the problem of face recognition were compared [6]. From these comparisons, the ArcFace [20] and DeepFace [18] models showed a great deal of potential as they yielded a very high accuracy score on large data-sets, with ArcFace giving an accuracy score of 99.83% on the MS-Celeb-1M data-set [6], [28] and DeepFace giving an accuracy score of 97.35% (±0.25) on Facebook's private data-set consisting of over 4404000 images [6]. Hence they were chosen to be the models of choice for O2.

To summarise the four chosen models that are going to be implemented in O2 are:

- Eigenfaces [10]
- Fisherface [11]
- ArcFace [20]
- DeepFace [18]

*1) **Eigenfaces**:* As previously explained, eigenfaces is classified as a holistic approach. Eigenfaces extracts the useful features from the set of all features by making use of PCA. PCA works by projecting a face onto other faces with the same identify. From this projection, common features can be extracted. This process is repeated until only the very core features are found [10]. This procedure can be clearly seen in Figure 2, where the features of a face that are deemed redundant are systemically removed.

Fig. 2. [29]

In order to successfully implement the PCA component of eigenfaces the scikit-learn library was used, specifically it's PCA decomposition component [30].

*2) **Fisherface**:* Fisherface is another holistic approach that is going to be used. Rather then making use of PCA to extract the useful features, it uses LDA [11]. This LDA technique works in a similar fashion to the PCA technique, but tends to be more lenient when choosing it's useful features. Due to this fact, fisherface tends to give better results on data-sets that have varying illumination conditions and images with various different facial expressions and poses [26].
To implement fisherface the OpenCV library was used, specifically it's FisherFaceRecognizer component [31]. As was expected, fisherface took longer to compute then eigenfaces, so to combat this the pixel values were converted to normalized pixels values.

*3) **ArcFace**:* ArcFace, also refereed to as additive angular margin loss, is a Deep Convolutional Neural Network (DCNN) that was built for the purpose of large scale face recognition.

What makes it stand out from other DCNN is it's loss function [20]. In deep learning the loss function evaluates how well a model is preforming at a given time period [32]. Based on the loss function's evaluation, changes are made in order to reduce the error rate of the model [32]. The most widely used classification loss function is the softmax loss function [20], [33], however in ArcFace a new loss function is proposed that outperforms not just the softmax loss function but other popular loss functions such as inter-loss and triplet-loss [20]. To implement the ArcFace an off-the-shelf implementation of ArcFace was used [34]. The python script follows the below pseudocode:

Algorithm 1: ArcFace pseudocode

```
// Variable Declaration
Initialize: machine learning model: M
Initialize: list of embeddings: trainingData
    // Calculated using ArcFace().calc_emb
Initialize: list of embeddings: testingData
    // Calculated using ArcFace().calc_emb
Initialize: int: found = 0
Initialize: int: blunder = 0

foreach element i ∈ testingData do
    Initialize: empty list: current
    foreach element j ∈ trainingData do
        Append the embedding distance between i and
            j to list current // Calculated using
            ArcFace().get_distance_embeddings
    end
    if i == min(current) then
        Increment found
    else
        Increment blunder
    end
end
// return accuracy score
Return: found / (found + blunder)
```

4) ***DeepFace:*** DeepFace is a deep neural network that has over 120 million parameters, that use locally connected layers without any weight-sharing [18]. DeepFace also splits up the face extraction process into two components, an align component and a represent component. In the alignment component specific points are extracting by making use of a support vector regressor. These extracted points are then first aligned on a 2D space, then converted to a 3D space [18]. This process is better highlighted in Figure 3.

To implement DeepFace a wrapper for Facebook's official DeepFace model was used [35]. All of the images had to first be passed through the pipeline shown in Figure 3. This procedure was very lengthy in terms of computational time, so the images were pre-processed and stored inside of a .pkl file.

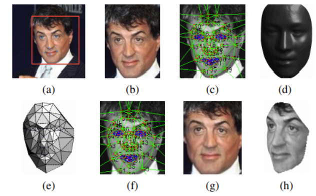

Fig. 3. Alignment pipeline used by DeepFace [18]

V. Evaluation

To properly evaluate the four models chosen in O2 the accuracy metric [36] is going to be used. Accuracy is the percentage of correctly made predictions. It represents the proportion of samples that are correctly classified and is defined as:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} = \frac{TP + TN}{TP + TN + FP + FN}$$

where:

- Accuracy ∈ [0, 1] (The bigger the better)
- TP = True positives
- TN = True negatives
- FP = False positives
- FN = False negatives

This metric was chosen over other metrics as it has been thoroughly used in past literature to measure the effectiveness of a given face recognition model [6], [18]. All of the four models were trained on 80% of the data and tested on the remaining 20%. Hence for every given person, 8 of their images were randomly chosen as training and the remaining 2 for testing. Since an element of randomness is involved all of the tests were carried out a total of five time in order to get a good average accuracy score.

In the table below one can see the results obtained from the four implemented models.

	ACC_1	ACC_2	ACC_3	ACC_4	ACC_5	ACC_A
ArcFace	1.0	0.9917	1.0	0.9917	0.9917	0.9950 (±0.0041)
DeepFace	0.9871	0.9615	0.9875	0.9875	0.9615	0.9770 (±0.0127)
Eigenfaces	0.975	0.95	0.9625	0.925	0.975	0.9575 (±0.0187)
Fisherface	0.975	0.975	0.975	1.0	0.9625	0.9775(±.0123)

As can be seen the ArcFace model achieved the highest accuracy with a near perfect score of 0.995. Surprisingly enough in second place there is a traditional model rather then a deep learning model. This was slightly unexpected as deep learning approaches tend to achieve better results. The slight drop in accuracy seen by the DeepFace model is probably due to the low amount of data that is available. As was expected the fisherface model performed better then the eigenfaces model. This is due to the fact that the data-set contains images under varying lighting conditions.

VI. Conclusion

Further work that can be done is to test these same models on even bigger data-sets, however this report showed how well these four models perform with a relatively small number of images per person, something which is common for many face recognition applications.

As has been highlighted in this report, face recognition has come a long way, with some models even claiming to have achieved near human level capabilities of face recognition [18]. Keeping this in mind, it should be interesting to see how these advancements in facial recognition that have happened over the past few years will be implemented in real world domains.

References

[1] R. Jafri and H. R. Arabnia, "A survey of face recognition techniques," vol. 5, no. 2, pp. 41–68, publisher: Korea Information Processing Society. [Online]. Available: https://www.koreascience.or.kr/article/JAKO200920237949770.page

[2] "Home | CVPR2020." [Online]. Available: http://cvpr2020.thecvf.com/

[3] "Cvpr2020 opening slides."

[4] D. Tao, Y. Guo, Y. Li, and X. Gao, "Tensor rank preserving discriminant analysis for facial recognition," vol. 27, no. 1, pp. 325–334.

[5] T. Huang, Z. Xiong, and Z. Zhang, "Face recognition applications," in *Handbook of Face Recognition*, S. Z. Li and A. K. Jain, Eds. Springer, pp. 371–390.

[6] I. Adjabi, A. Ouahabi, A. Benzaoui, and A. Taleb-Ahmed, "Past, Present, and Future of Face Recognition: A Review," Jul. 2020, publisher: Preprints. [Online]. Available: https://www.preprints.org/manuscript/202007.0479/v1

[7] E. Hjelmås and B. K. Low, "Face Detection: A Survey," *Computer Vision and Image Understanding*, vol. 83, no. 3, pp. 236–274, Sep. 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S107731420190921X

[8] V. Lempitsky, P. Kohli, C. Rother, and T. Sharp, "Image segmentation with a bounding box prior," in *2009 IEEE 12th International Conference on Computer Vision*, Sep. 2009, pp. 277–284, iSSN: 2380-7504.

[9] X. Cao, X. Wei, Y. Han, and D. Lin, "Robust Face Clustering Via Tensor Decomposition," *IEEE Transactions on Cybernetics*, vol. 45, no. 11, pp. 2546–2557, Nov. 2015, conference Name: IEEE Transactions on Cybernetics.

[10] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, Jan. 1991. [Online]. Available: https://doi.org/10.1162/jocn.1991.3.1.71

[11] M. Anggo and L. Arapu, "Face Recognition Using Fisherface Method," *Journal of Physics: Conference Series*, vol. 1028, p. 012119, Jun. 2018.

[12] A. Tharwat, "Independent Component Analysis: an Introduction," *Applied Computing and Informatics*, vol. ahead-of-print, Aug. 2018.

[13] F. Y. Shih and C.-F. Chuang, "Automatic extraction of head and face boundaries and facial features," *Information Sciences*, vol. 158, pp. 117–130, Jan. 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S002002550300197X

[14] C. Kotropoulos, A. Tefas, and I. Pitas, "Frontal face authentication using morphological elastic graph matching," *Image Processing, IEEE Transactions on*, vol. 9, pp. 555–560, May 2000.

[15] D. Kumar, J. Garain, D. R. Kisku, J. K. Sing, and P. Gupta, "Unconstrained and constrained face recognition using dense local descriptor with ensemble framework," *Neurocomputing*, vol. 408, pp. 273–284, Sep. 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231220304914

[16] T. Ahonen, A. Hadid, and M. Pietikäinen, *Face Recognition with Local Binary Patterns*, May 2004, vol. 3021, pages: 481.

[17] J. Kannala and E. Rahtu, "BSIF: Binarized statistical image features," Nov. 2012, pp. 1363–1366, iSSN: 1051-4651.

[18] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. Columbus, OH, USA: IEEE, Jun. 2014, pp. 1701–1708. [Online]. Available: https://ieeexplore.ieee.org/document/6909616

[19] Y. Liu, H. Li, and X. Wang, "Rethinking Feature Discrimination and Polymerization for Large-scale Recognition," *arXiv:1710.00870 [cs]*, Oct. 2017, arXiv: 1710.00870. [Online]. Available: http://arxiv.org/abs/1710.00870

[20] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," *arXiv:1801.07698 [cs]*, Feb. 2019, arXiv: 1801.07698. [Online]. Available: http://arxiv.org/abs/1801.07698

[21] F. Samaria and A. Harter, *Parameterisation of a stochastic model for human face identification*, Jan. 1995, vol. 22, journal Abbreviation: IEEE Workshop on Applications of Computer Vision - Proceedings Pages: 142 Publication Title: IEEE Workshop on Applications of Computer Vision - Proceedings.

[22] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.

[23] S. Sengupta, J.-C. Chen, C. Castillo, V. M. Patel, R. Chellappa, and D. W. Jacobs, "Frontal to profile face verification in the wild," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. Lake Placid, NY, USA: IEEE, Mar. 2016, pp. 1–9. [Online]. Available: http://ieeexplore.ieee.org/document/7477558/

[24] G. van Rossum, "Python tutorial," Centrum voor Wiskunde en Informatica (CWI), Amsterdam, Tech. Rep. CS-R9526, May 1995.

[25] D. Radečić, "daradecic/Python-Eigenfaces," Nov. 2020, original-date: 2020-05-27T10:33:01Z. [Online]. Available: https://github.com/daradecic/Python-Eigenfaces

[26] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, Jul. 1997, conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[27] Karahan and Y. S. Akgül, "Eye detection by using deep learning," May 2016, pp. 2145–2148.

[28] J. Harvey, Adam. LaPlace. (2019-2020) Megapixels: Origins and endpoints of biometric datasets "in the wild". [Online]. Available: https://megapixels.cc

[29] M. Turk, "Eigenfaces." [Online]. Available: http://www.scholarpedia.org/article/File:Eigenfaces.jpg

[30] M. B. D. A. E. M. E. G. P. Alexandre Gramfort, Olivier Grisel, "scikit-learn/scikit-learn." [Online]. Available: https://github.com/scikit-learn/scikit-learn

[31] "opencv/opencv," Jan. 2021, original-date: 2012-07-19T09:40:17Z. [Online]. Available: https://github.com/opencv/opencv

[32] L. Wu, Z. Zhu, and W. E, "Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes," *arXiv:1706.10239 [cs, stat]*, Nov. 2017, arXiv: 1706.10239. [Online]. Available: http://arxiv.org/abs/1706.10239

[33] K. Janocha and W. M. Czarnecki, "On Loss Functions for Deep Neural Networks in Classification," *arXiv:1702.05659 [cs]*, Feb. 2017, arXiv: 1702.05659. [Online]. Available: http://arxiv.org/abs/1702.05659

[34] P. Hofer, "arcface: ArcFace face recognition implementation in Tensorflow Light." [Online]. Available: https://github.com/mobilesec/arcface-tensorflowlight

[35] S. I. Serengil, "serengil/deepface," Jan. 2021, original-date: 2020-02-08T20:42:28Z. [Online]. Available: https://github.com/serengil/deepface

[36] *ISO 5725-1 Accuracy (trueness and precision) of measurement methods and results - Part 1: General principles and definitions*. International Organization for Standardization, Dec. 1994.