Finding Similar non-Collapsed Faces to Collapsed Faces Using Deep Learning Face Recognition

Ashwinee Mehta[1][a], Maged Abdelaal[2][b], Moamen Sheba[2][c] and Nic Herndon[1][d]

[1]*Department of Computer Science, East Carolina University, Greenville, U.S.A.*

[2]*School of Dental Medicine, East Carolina University, Greenville, U.S.A.*

Keywords: Similar, non-Collapsed Face, Face Recognition, Classification, Collapsed Face, Reconstruction.

Abstract: Face recognition is the ability to recognize a person's face in a digital image. Common uses of face recognition include identity verification, automatically organizing raw photo libraries by person, tracking a specific person, counting unique people and finding people with similar appearances. However, there is no systematic and accurate study for finding a similar non-collapsed face to a given collapsed face. In this paper we focus on the use case of finding people with similar appearances that will help us to find a similar face without a collapse to a collapsed face for dental reconstruction. We used Python's Open-CV for age and gender classification and face recognition for finding similar faces. Our results provide a set of similar images that can be used for reconstructing the collapsed faces for creating dentures. Thus with the help of a similar non-collapsed face, we can reconstruct a collapsed face for designing effective dentures.

1 INTRODUCTION

Images contain a lot of crucial information that can be used for a variety of applications. The importance of search applications that closely match facial features in image-based searches is increasing day by day. We all have wondered if there was someone out there who looked just like us. We are all aware of the urban legend that there are six people out there who look just like us. The problem lies in finding those similar people. Face matching and retrieval can be used in forensics applications in matching forensic sketches to face photograph databases. Face matching also has its applications in recommender systems for glasses, hairstyle, jewellery, etc., and in dentistry for finding a similar face for facial reconstruction and denture designing.

A lot of work has been performed in the field of facial similarity where different methods and metrics have been proposed for finding similar faces to a given input face like Scale-Invariant Feature Transform (SIFT) descriptors, Lookalike networks, Local Binary Pattern (LBP), Doppelganger lists, etc. There are situations in real life where it is required to find a similar face to a face which is collapsed or has structural deformities. However, none of the existing methods have focused on faces with deformities or a collapse. All the existing methods have the input and output similar images of normal faces without collapse or deformities where the facial similarity is calculated and a similar face is returned from a target dataset.

Missing human teeth cause the body to reabsorb the bone that supported the teeth. Over the course of about 10-20 years the jawbone shrinks significantly which results in a condition known as facial collapse. Humans with facial collapse appear much older than they are. The facial collapse not only alters the human's facial appearance, but also affects their dental health. This facial collapse can be prevented with the placement of dental implants. The implant sends a piezoelectric signal to the bone which prevents the bone from reabsorption. Dentures are designed by using the denture impressions of the jaw and mouth after which the dentist will create models usually from wax or plastic, based off the impression. The patient then tries the model several times to check for fit, shape, and even color before the denture is made. The current denture design workflow does not have a systematic approach to include the aesthetic factors, patient's pre-treatment facial shape and in-progress

[a] https://orcid.org/0000-0002-7167-2563
[b] https://orcid.org/0000-0002-7414-423X
[c] https://orcid.org/0000-0003-1188-2080
[d] https://orcid.org/0000-0001-9712-148X

Mehta, A., Abdelaal, M., Sheba, M. and Herndon, N.
Finding Similar non-Collapsed Faces to Collapsed Faces Using Deep Learning Face Recognition.
DOI: 10.5220/0011696900003417
In *Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2023) - Volume 4: VISAPP*, pages 897-904
ISBN: 978-989-758-634-7; ISSN: 2184-4321

denture design visualizations, instead relying on discussing mockups with the patients during appointments. This results into waiting for the final denture fitting on the patient to evaluate the final denture aesthetics.

With a collapsed face, only the bottom one third of the face is affected and needs to be restored before denture designing. This facial restoration needs a similar face which can be used as a reference for comparing the facial shape proportions to convert the collapsed face into non-collapsed facial shape. Inclusion and evaluation of facial aesthetics is important while planning for facial or dental reconstruction treatment.

Our proposed method focuses towards the goal of finding a similar non-collapsed face to a given collapsed face for reconstructing the lower third of the collapsed face before designing the denture. We have used Python's Deep Learning Face Recognition (Geitgey, 2018) which is built using Dlib (King, 2009) for finding similar non-collapsed faces to a given collapsed face from a diverse target dataset of human population. By using the proposed method, one can find multiple similar images and use them for facial reconstruction for automatic denture designing.

2 MATERIALS AND METHODS

We have worked on finding similar non-collapsed faces for a given collapsed face by using the Human Faces dataset (kaggle, 2020). The Human Faces Dataset contains 224,500 images of human faces. The image data in the dataset has been generated using StyleGAN2 (Karras et al., 2020) which focuses on improving the resolution and quality of images. The StyleGAN2 was trained by using the Flickr Faces HQ (FFHQ) and Large-scale Scene UNderstanding Challenge (LSUN) datasets. The Human Faces Dataset contains nine directories and each such directory contains multiple folders that contain the image files. Each image file contains an image of a single human face. The dataset is diverse in a way that it contains images of human faces of different age groups beginning from toddlers to old humans and different genders, i.e., male and female, yet there's no metadata (e.g., age, race, etc.). All the image files in the dataset have a resolution of 1024×1024.

OpenCV's gender and age classification is based on a convolutional neural network architecture with a total of 3 convolutional layers, 2 fully connected layers and a final output layer. "Conv1" is the first convolutional layer that has 96 nodes of kernel size 7. "Conv2" is the second convolutional layer that has 256 nodes of kernel size 5. "Conv3" is the third convolutional layer that has 384 nodes of kernel size 3. The two fully connected layers have 512 nodes each. Gender prediction is framed as a classification problem. The output layer in the gender prediction network is of type softmax with 2 nodes indicating the two classes "Male" and "Female". Age Prediction should be approached as a regression problem since we are expecting a real number as the output. However, it is difficult to accurately estimate the age of a person and even humans find it challenging. Hence, age prediction was framed as a classification problem where we try to estimate the age group the person is in. The age prediction network has 8 nodes in the final softmax layer indicating the age ranges 0 to 2, 4 to 6, 8 to 12, 15 to 20, 25 to 32, 38 to 43, 48 to 53 and 60 to 100 years old. The different model files used and loaded for performing the age and gender detection task are listed below:

- **`gender_net.caffemodel`**: It is the pre-trained model weights for gender detection.
- **`deploy_gender.prototxt`**: It is the model architecture for the gender detection model.
- **`age_net.caffemodel`**: It is the pre-trained model weights for age detection.
- **`deploy_age.prototxt`**: It is the model architecture for the age detection model.
- **`res10_300x300_ssd_iter_140000_fp16.caffemodel`**: It is the pre-trained model weights for face detection.
- **`deploy.prototxt.txt`**: It is the model architecture for the face detection model.

Our proposed workflow as shown in Figure 2 had the following steps:

1. Image Classification: In order to find a similar non-collapsed face, it was important to find the similar face around the same age-group and gender. The first step was to classify the image according to gender into male and female categories. We used Python's OpenCV for age and gender detection of human faces in the image files. The images were classified into two genders, i.e., male and female, and further classified into different age groups under each gender. The different age classes included 0 to 2, 4 to 6, 8 to 12, 15 to 20, 25 to 32, 38 to 43, 48 to 53 and 60 to 100 years old.

The different steps involved in classification of gender and age include:

1. Read the image using the `cv2.imread()` method.
2. After the image is resized to the appropriate size, use the `get_faces()` function to get all the detected faces from the image.

3. Iterate on each detected face image and call our `get_age_predictions()` and `get_gender_predictions()` to get the predictions.
4. Print the age and gender along with the confidence levels.

2. Generating Image File Metadata: After detection of gender and age, the next step was to store the the output of classification as metadata for every image file. The file name of every image file was renamed to include the metadata of age and gender and their respective confidence levels.

3. Sorting Image Files Into Folders: After generating the image file metadata, the next step was to sort the classified data into different folders based on the gender and age for searching the similar non-collapsed face to a collapsed face in the folder that corresponds to the collapsed face's detected gender and age (e.g., `male_0to2`, `male_4to6`, etc.).

4. Generate Similar Images: After having all the images sorted in different folders as per their classification result, the next step was to look for a similar non-collapsed face to the given input collapsed face based on the gender and age of the input collapsed face by using Python's Deep Learning Face Recognition as shown in Figure 1. We found out that the process of generating similar images is slow. To speed up the process we created small samples of each folder mentioned in Step 3 that contained 500 images and generated similar images by searching only the sample folder.

3 RESULTS AND DISCUSSION

Figure 3 shows the query image that contains a collapsed face of a male in the age group 60 to 100 and the top ten similar images to the query image ranked by their similarity scores. Figure 6 shows the query image of a collapsed face of a female in the age group 48 to 53 and the top ten similar images to the query image ranked by their similarity scores. In Figure 3, the most similar face has a similarity score of 0.6050 followed by the other less similar faces with an increasing similarity scores. In Figure 6, the most similar face has a similarity score of 0.6210 followed by other less similar faces with an increasing similarity scores. Out of all the images involved in the experiment, we can say that, using the proposed method, the image with the lowest score is the most similar image to the given query image. When we visually evaluate the results, we can confirm that this is the case for most of the query images.

In Figure 3, Figure 5 and Figure 6, the gender of all the similar images in the output matches the gender of the query image that contains a collapsed face. In Figure 3 the query image is of a male and all the similar images are male and in Figures 5 and 6, the query images are of a female and all the similar images are female. Also, the age of most of the similar images matches with the age of the query image in Figure 3 and Figure 5. Having ten similar images gives the flexibility to choose from the set that might have some images that do not satisfy some criteria. The proposed method also works for faces in all age groups that do not have a collapsed face.

The dataset had some duplicate images which were returned in the similarity results. In this situation, only one such image was selected and the next similar image was considered. Some images in the dataset did not have clear face visibility due to which there were errors. After excluding such images, we were able to find similar non-collapsed images to a given collapsed face. Since the dataset used in this paper did not have labelled data, we used machine learning to classify the images into different age and gender categories. During this process of classification, there were images that were not classified into correct folders. Thus some folders contained images that belong to other age and gender category. This resulted into missing out on some images that were potential candidates for similar images. During the step of generating image file metadata i.e., gender and age labels, Python's OpenCV based age and gender detection also displays a confidence score for each label. This confidence score can be used for filtering out incorrectly labelled images from the age and gender category folders so that each folder contains only correctly labelled images.

4 RELATED WORK

In (Torun et al., 2009), a method is developed to match similar faces in scattered datasets and to recognize images of a person in different states. A solution that works more effectively than traditional face recognition methods, which works with a high error rate according to different exposure values, is presented. By using SIFT descriptors that are used in object recognition problems, first, the face's points of interest are identified followed by matching points of interests between two faces. Then, the similarity ratios of two faces to each other are calculated by looking at the distances. The average distances of the points of interest assumed to be correctly mapped determined this ratio.

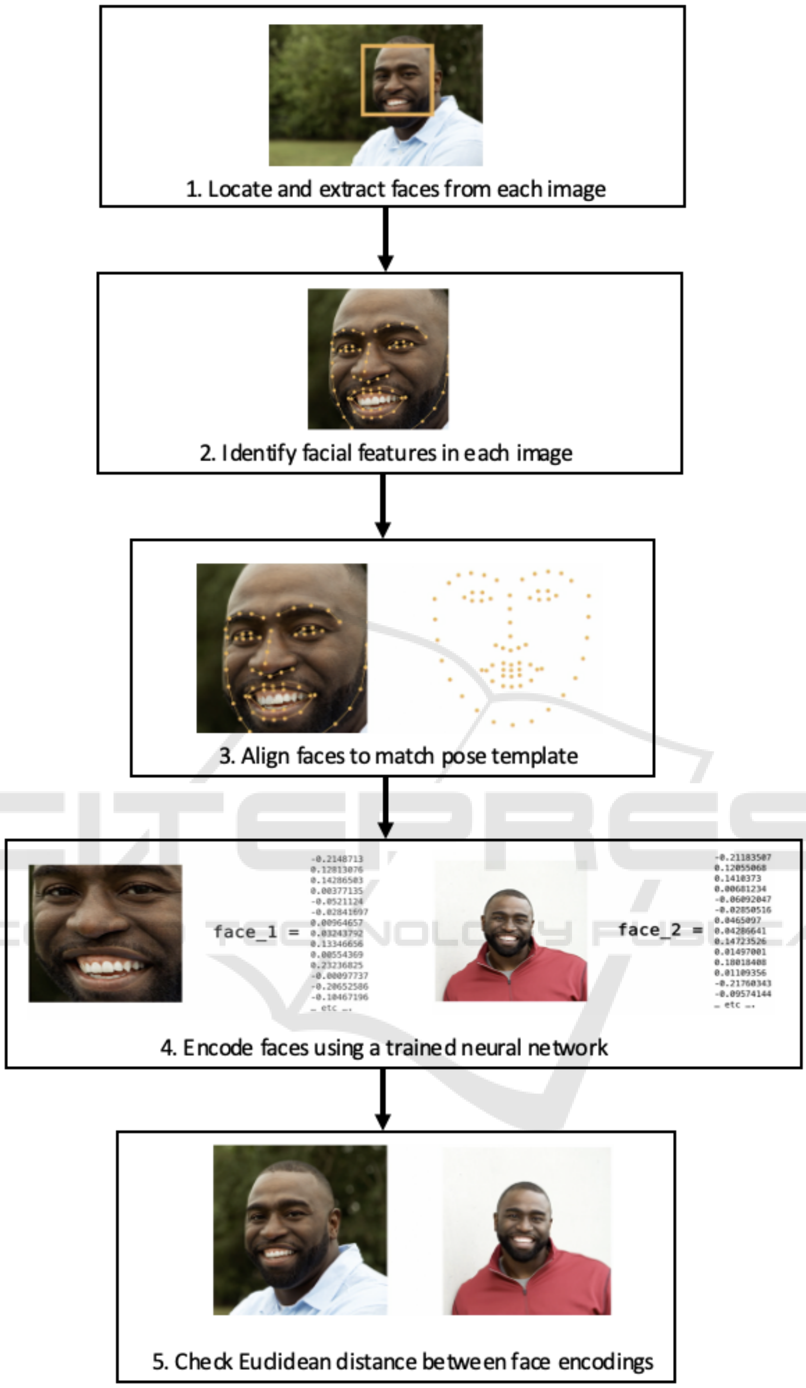

Figure 1: Face Recognition Pipeline.

In (Schroff et al., 2011) a new method is introduced for comparing pairs of face images that allows recognition or verification between images where the mutually visible portion of face is small. They have explored the idea of using a sorted Doppelganger list as a signature and evaluate distance functions for comparing the signatures. Each probe in a query pair is compared to all members of the Library. The comparison results in a ranked list of look-alikes, the first being the most similar to the query. Then, a similarity measure between the two probes is computed by comparing the ranked lists.

In (Sadovnik et al., 2018), the emphasis is on presenting evidence that finding facial look-alikes and

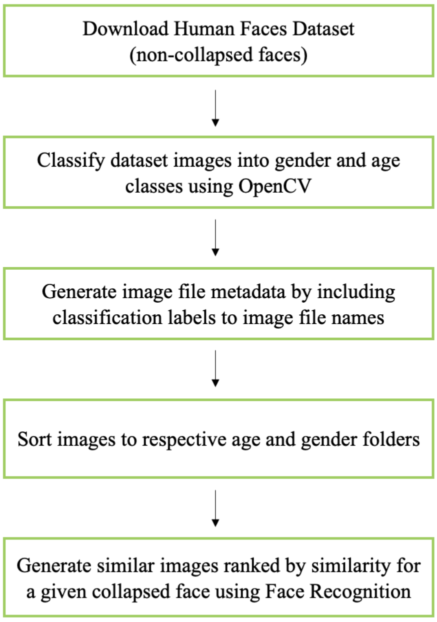

Figure 2: Proposed Workflow.

recognizing faces are two distinct tasks. They expected that many features which are useful for face recognition will also be useful for face similarity. They have used the pre-trained VGG face CNN descriptor network for face-recognition and then fine-tuned the weights on a new dataset that is targeted at capturing preceived facial similarity to perform well at the face similarity task by proposing the lookalike network. The results show that the proposed method outperforms the face recognition baseline at the task of predicting which faces will appear more similar to human.

In (Ramanathan et al., 2004), the focus is on deriving a measure of similarity between faces. They mention that illumination, pose variations, disguises, aging effects and expression variations are some of the key factors that affect the performance of face recognition systems. They have suggested a framework to compensate for pose variations and introduce the notion of 'Half-faces' to circumvent the problem of non-uniform illumination and used the similarity measure to retrieve similar faces from a database containing multiple images of individuals. They concluded that the similarity measure helps in studying the significance facial features play in affecting the performance of face recognition systems.

In (McCauley et al., 2021a), they have used the twin database to calculate a baseline measure of the worst-case scenario of facial similarity in Face Recognition using a deep CNN. They have carried out a performance analysis of two Face Recognition tools presented with highly similar faces and using an experimental twin threshold, potential look-alikes were extracted from the datasets for further analysis. The similarity measure presented in this paper can be used to compare facial similarity to a comparison score from a Face Recognition system in order to better understand the impact that facial similarity has on Face Recognition and to identify potential look-alikes from large datasets.

In (Alasadi et al., 2019), they have showed that adopting an adversarial deep learning-based approach allows for the model to maintain the accuracy at face matching while also reducing demographic disparities compared to a baseline (non-adversarial deep learning) approach at face matching and paves way for more accurate and fair face matching algorithms. They have proposed a deep-learning adversarial approach for reducing bias in face-matching algorithms. The proposed GAN-based framework consisted of two parts- one which tries to maximize the face matching quality and the other which tries to minimize the ability of the network to infer the demographic properties of the individual whose facial image is under consideration.

In (Bicego et al., 2005), they have presented a novel approach for extracting characteristic parts of a face. Instead of finding a priori specified features such as nose, eyes, mouth or others, the proposed approach is aimed at extracting from a face the most distinguishing or dissimilar parts with respect to another given face, i.e. at "finding differences" between faces by feeding a binary classifier by a set of image patches, randomly sampled from the two face images, and scoring the features by their mutual distances.

In (Ruys et al., 2006), they have investigated the role of differences between people in the comparison process. They have proposed a dual process model in which dissimilarity can function in two different ways with opposite effects on social judgement. They have discussed two different arguments- 1. Dissimilarity between people may decrease the likelihood of placing them in the same category. 2. Activated dissimilarity may determine the detection of feature overlap between people during comparison and therefore influence the holistic similarity assessment.

In (O'Toole et al., 2007), they have compared seven state-of-the-art face recognition algorithms with humans on a facematching task where humans and algorithms determined whether pairs of face images, taken under different illumination conditions, were pictures of the same person or of different people. It was found out that three algorithms surpassed human performance matching "difficult" face pairs and six algorithms surpassed humans on "easy" face pairs. They mention that although illumination vari-

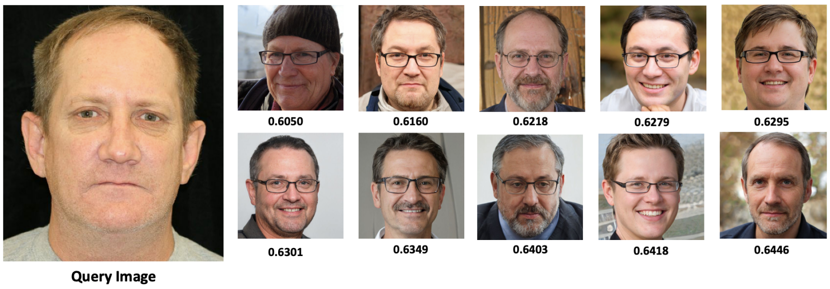

Figure 3: The query image is a collapsed face of a male in the age group 60 to 100. Top ten non-collapsed faces ranked by their similarity are found that can be used for reconstructing the bottom third of the collapsed face. The next two ranked images has the similarity of 0.7569 and 0.7574.

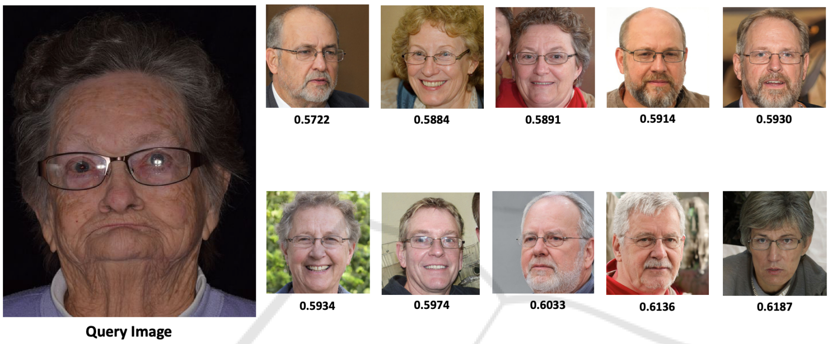

Figure 4: The query image is a collapsed face of a female in the age group 60 to 100. Top ten non-collapsed faces ranked by their similarity are found that can be used for reconstructing the bottom third of the collapsed face. The next two ranked images has the similarity of 0.6223 and 0.6243.

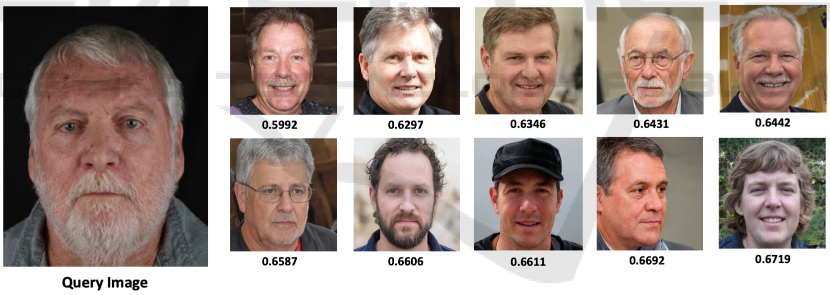

Figure 5: The query image is a collapsed face of a male in the age group 48 to 53. Top ten non-collapsed faces ranked by their similarity are found that can be used for reconstructing the bottom third of the collapsed face. The next two ranked images has the similarity of 0.7033 and 0.7079.

ation continues to challenge face recognition algorithms, current algorithms compete favorably with humans.

(Kramer and Reynolds, 2018) have focused on face matching using profile images. They have compared face matching accuracy when both frontal and profile image of each face were presented, with accuracy using each view alone. Surprisingly, they found no benefit when both views were presented together. Also, they found out that there was no difference in performance when front and profile views were used suggesting that both views were similarly useful for face matching. Overall, these results suggest that either frontal or profile views provide substantially overlapping information regarding identity or participants are unable to utilize both sources of information when making decisions.

In (Oron et al., 2018), they have proposed a new method called Best-Buddies Similarity for template matching using mutual nearest neighbors. The proposed method follows the traditional sliding window approach and by computing the Best-Buddies Similarity between the template and every window of the size of the template in the image. Best-Buddies Similarity is calculated by leveraging statistical properties of mutual nearest neighbors and was shown to be use-

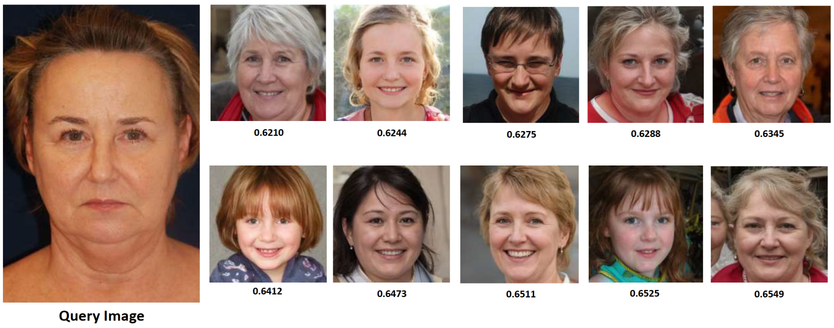

Figure 6: The query image is a collapsed face of a female in the age group 48 to 53. Top ten non-collapsed faces ranked by their similarity are found that can be used for reconstructing the bottom third of the collapsed face. The next two ranked images has the similarity of 0.65763 and 0.65766.

ful for template matching in the wild. Key features of BBS were identified and analyzed demonstrating its ability to overcome several challenges that are common in real life template matching scenarios. They also mention that the proposed method might have additional applications in computer-vision or other fields that could benefit from its properties.

In (Santini and Jain, 1996), focus is on the problem of similarity matching. They state that similarity matching is the single most important operation in image databases. They have discussed some of the models of human similarity like Euclidean Distance, City-Block Distance, Thurstone-Shepard Models, the Feature Contrast Model and they Fuzzy Feature Contrast Model. They have used images from the MIT face images dataset for measuring similarity based on geometric measures. They have modified the Feature Contrast Model into the Fuzzy Feature Contrast Model. From the experimental results, they prove that the Fuzzy Feature Contrast Model is almost symmetric in case of human faces.

In (McCauley et al., 2021b), they have discussed the problem of distinguishing identical twins and non-twin look-alikes. They address two challenges- determining a baseline measure of facial similarity between identical twins and applying this similarity measure to determine the impact of doppelgangers, or look-alikes, on FR performance for large face datasets. The facial similarity measure is determined via a deep convolutional neural network. The proposed network provides a quantitative similarity score for any two given faces and has been applied to large-scale face datasets to identify similar face pairs.

In (Röttcher et al., 2020), they have proposed a new method for efficiently select similar pairs. They have compared the proposed method with the adapted version of a random selection process which is often found in state-of-the-art morphing attack research. The conducted experiment proves that appropriate pair selection not only increases the morph quality but also substantially decreases the standard deviation between many morphing techniques. An effective pre-selection reduces the need for a perfect, low-artefact producing, morphing algorithm. This is important as automated morphing is still error-prone with the difficulty to fully remove all the artefacts. Automated face recognition systems that operate on the purpose of determining the similarity between two facial images are not only vulnerable to morphed faces but they can also contribute to a morphing attack by finding optimal pairs of data subjects in a sufficient manner.

In (Lamba et al., 2011), they have prepared a look-alike database and analyzed human performance with the help of 50 volunteers. They have used both subspace and texture descriptor based algorithms for automatic algorithms. Their experimental results suggest that, for look-alikes, humans and automatic algorithms do not perform better than random guess. They have also also proposed an algorithm that significantly improves the performance compared to existing algorithms. They state that it is important to start considering complex covariates including look-alikes and develop advanced algorithms.

5 CONCLUSIONS AND FUTURE WORK

The bottom third of the face is adversely affected when a facial collapse due to missing teeth and jaw bone loss occurs. It is important to properly restore this third of the face before designing dentures. The lower third of a collapsed face can be restored by using a similar non-collapsed face as a reference face. We have proposed a method to find a similar non-collapsed face to a given collapsed face using Python's Deep Learning Face Recognition. Our results provide the similar non-collapsed images to the given collapsed image which can be used for reconstructing the bottom third of a collapsed face before designing dentures.

When looking for a similar non-collapsed face which can be used as a reference to the query image that contains a collapsed face, it would be ideal to have a reference face that is a closest match to the query image as this would help in more accurate reconstruction procedure of the query image that contains a collapsed face. Since this proposed method is about finding similar images from the target dataset, the bigger and diverse the dataset is, the more is the probability of finding a closer and a similar match to the query image. However, it should be made sure that the target dataset does not contain images of people with a collapsed face. In addition to age and gender, there can be other factors like race, ethnicity, skin color, height, weight, etc. that can affect the similarity score and help to find the closest match. It is worth exploring the similar images in a more wide and diverse dataset to check for the similarity scores and similar images. Also, we would like to collect images of patients that have collapsed faces to evaluate the proposed method.

REFERENCES

Alasadi, J., Al Hilli, A., and Singh, V. K. (2019). Toward fairness in face matching algorithms. In *Proceedings of the 1st International Workshop on Fairness, Accountability, and Transparency in MultiMedia*, FAT/MM '19, page 19–25, New York, NY, USA. Association for Computing Machinery.

Bicego, M., Grosso, E., and Tistarelli, M. (2005). On finding differences between faces. In *International Conference on Audio-and Video-Based Biometric Person Authentication*, pages 329–338. Springer.

Geitgey, A. (2018). Face recognition library.

kaggle (2020). Human faces dataset.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119.

King, D. E. (2009). Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758.

Kramer, R. S. and Reynolds, M. G. (2018). Unfamiliar face matching with frontal and profile views. *Perception*, 47(4):414–431.

Lamba, H., Sarkar, A., Vatsa, M., Singh, R., and Noore, A. (2011). Face recognition for look-alikes: A preliminary study. In *2011 International Joint Conference on Biometrics (IJCB)*, pages 1–6.

McCauley, J., Soleymani, S., Williams, B., Dando, J., Nasrabadi, N., and Dawson, J. (2021a). Identical twins as a facial similarity benchmark for human facial recognition. In *2021 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–5.

McCauley, J., Soleymani, S., Williams, B., Dando, J., Nasrabadi, N., and Dawson, J. (2021b). Identical twins as a facial similarity benchmark for human facial recognition. In *2021 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–5.

Oron, S., Dekel, T., Xue, T., Freeman, W. T., and Avidan, S. (2018). Best-buddies similarity—robust template matching using mutual nearest neighbors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(8):1799–1813.

O'Toole, A. J., Jonathon Phillips, P., Jiang, F., Ayyad, J., Penard, N., and Abdi, H. (2007). Face recognition algorithms surpass humans matching faces over changes in illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1642–1646.

Ramanathan, N., Chellappa, R., and Roy Chowdhury, A. (2004). Facial similarity across age, disguise, illumination and pose. In *2004 International Conference on Image Processing, 2004. ICIP '04.*, volume 3, pages 1999–2002 Vol. 3.

Röttcher, A., Scherhag, U., and Busch, C. (2020). Finding the suitable doppelgänger for a face morphing attack. In *2020 IEEE international joint conference on biometrics (IJCB)*, pages 1–7. IEEE.

Ruys, K. I., Spears, R., Gordijn, E. H., and de Vries, N. K. (2006). Two faces of (dis) similarity in affective judgments of persons: Contrast or assimilation effects revealed by morphs. *Journal of Personality and Social Psychology*, 90(3):399.

Sadovnik, A., Gharbi, W., Vu, T., and Gallagher, A. (2018). Finding your lookalike: Measuring face similarity rather than face identity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Santini, S. and Jain, R. (1996). Similarity matching. In *Asian conference on computer vision*, pages 571–580. Springer.

Schroff, F., Treibitz, T., Kriegman, D., and Belongie, S. (2011). Pose, illumination and expression invariant pairwise face-similarity measure via doppelgänger list comparison. In *2011 International Conference on Computer Vision*, pages 2494–2501.

Torun, R. B., Yurdakul, M., and Duygulu, P. (2009). Finding similar faces. In *2009 IEEE 17th Signal Processing and Communications Applications Conference*, pages 960–963.