Texture-based 3D Face Recognition using Deep Neural Networks for Unconstrained Human-machine Interaction

Michael Danner[1,2], Patrik Huber[1,3], Muhammad Awais[1], Zhen-Hua Feng[1], Josef Kittler[1]
and Matthias Raetsch[2]

[1] *Centre for Vision, Speech & Signal Processing, University of Surrey, Guildford, U.K.*
[2] *ViSiR, Reutlingen University, Reutlingen, Germany*
[3] *Department of Computer Science, University of York, York, U.K.*

Keywords: Face Recognition, Deep Learning, 3D Morphable Face Model, 3D Reconstruction.

Abstract: 3D assisted 2D face recognition involves the process of reconstructing 3D faces from 2D images and solving the problem of face recognition in 3D. To facilitate the use of deep neural networks, a 3D face, normally represented as a 3D mesh of vertices and its corresponding surface texture, is remapped to image-like square isomaps by a conformal mapping. Based on previous work, we assume that face recognition benefits more from texture. In this work, we focus on the surface texture and its discriminatory information content for recognition purposes. Our approach is to prepare a 3D mesh, the corresponding surface texture and the original 2D image as triple input for the recognition network, to show that 3D data is useful for face recognition. Texture enhancement methods to control the texture fusion process are introduced and we adapt data augmentation methods. Our results show that texture-map-based face recognition can not only compete with state-of-the-art systems under the same preconditions but also outperforms standard 2D methods from recent years.

1 INTRODUCTION

Recent developments in deep Convolutional Neural Networks (CNNs) led to significant advancements in the field of face recognition. For the case of frontal face recognition, deep learning systems are already outperforming humans. But recognition of non near-frontal faces under uncontrolled imaging and illumination conditions still remains a challenge. The state-of-the-art recognition systems mostly treat faces as 2D objects. These systems detect faces in an image and then apply some kind of geometric transformation like rotation and translation on the 2D images, and feed the resulting images as input to a CNN. We will extend this input, as human faces are in reality 3D objects and consist of 3D shape and skin texture. There are some existing face recognition systems (Masi et al., 2016; Kittler et al., 2018; Koppen et al., 2018) which take 3D shape of face into account and synthesise frontal faces by using 3D information. Kittler *et al.* (Kittler et al., 2018) have combined 3D shape and texture information for face recognition and have shown that the system benefits more from texture than from shape. Motivated by their work, in this paper we investigate different approaches to improve a face recognition system based on texture maps. We fit a 3D face model to 2D input images in the wild in order to recover 3D shape and surface texture. We then transform 3D data to a 2D output image by storing pose information in *RGBA* images, where the alpha-values represent the view-angle of the face to the camera for each pixel. The output of merging texture maps are not a beauty contest for human eyes to produce smooth facial texture but to augment the dataset and provide the deep learning network with discriminative information.

Using regular 2D images as input and output of our preprocessing pipeline enables us to use common face recognition datasets and allows us to exploit latest CNN architectures. Last, we employ data augmentation for training deep neural networks. We generate additional training data by merging and blending face textures of one subject from multiple images using a quality-controlled fusion. We show that, with the proposed face texture enhancements, our face recognition pipeline achieves a performance comparable to state-of-the-art 2D face recognition methods.

Figure 1 depicts the pipeline of our 3D assisted 2D face recognition system that has two main compo-

Danner, M., Huber, P., Awais, M., Feng, Z., Kittler, J. and Raetsch, M.
Texture-based 3D Face Recognition using Deep Neural Networks for Unconstrained Human-machine Interaction.
DOI: 10.5220/0008982504200427
In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020) - Volume 5: VISAPP*, pages 420-427
ISBN: 978-989-758-402-2; ISSN: 2184-4321

nents: a) 3D assisted data processing, b) face recognition engine based on 2D input. In this processing step, we perform face and facial landmark detection for a 2D image in the wild. We then reconstruct the 3D face of an input 2D image by recovering its 3D shape and skin texture. This is achieved by fitting a 3D face model using 2D facial landmarks. The surface face texture of the input image is remapped onto a 2D isomap projection of the 3D mesh, to produce texture maps. We choose to work with a rectangular isomap projection that fills the whole space with pixel information, so that it is especially amenable to convolutional processing. Every texture map is then analysed, rated, enhanced and augmented to train deep CNNs.

The main goal of texture improvement is to enhance discriminatory information of skin texture of 3D faces by combining cues from multiple training images. This leads to the following contributions of the papers. First, to address shape distortions in isomaps resulting from imprecise fittings, we propose geometric correction of texture maps. Second, to fill missing texture values in isomaps due to large non-frontal poses we propose texture merging. For a smooth texture merging between different training images of the same subject we propose a measure of texture quality. Last, data augmentation plays a significant role in training of CNNs, therefore, to train deep CNN for face recognition we propose a 3D data augmentation technique. For effective data augmentation we utilised proposed texture quality measure to merge and blend face textures of one subject from multiple images. In the progress of this work we create four variants of the CASIA dataset to enhance training, which consist of (a) the unedited 2D skin texture maps, (b) merged texture maps by an alpha compositing algorithm, (c) merged texture maps by a Poisson blending algorithm and (d) a dataset where the original texture maps are aligned, warped and enhanced.

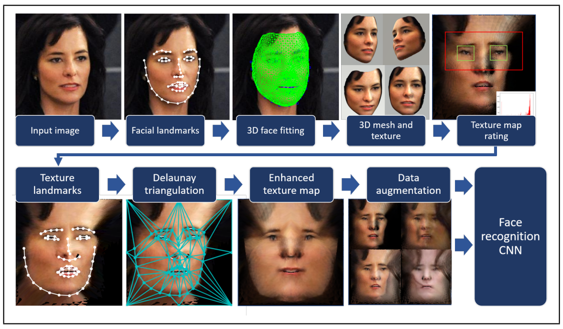

Figure 1: The proposed 3D assisted 2D face recognition pipeline.

The face recognition training is evaluated on the LFW (Huang et al., 2007) and IJB-A (Klare et al., 2015) datasets and compared to conventional 2D face recognition results. We carry out training on each of the four texture map datasets using an Inception-ResNet-v1 (Szegedy et al., 2017) architecture and achieve competitive results. This has exceeded our expectations greatly because extracting texture and using a 2D isomap representation have little impact on losing discriminative information for face recognition tasks. Additionally, the consolidation of the four texture map datasets for an augmented training matches state-of-the-art performance.

The remainder of this paper is structured as follows: In Section 1, we discuss related work on the development of face analysis, deep learning and 3D assisted face recognition. In Section 2, we show an overview of the proposed framework and introduce our methodology including texture map rating, reconstruction and augmentation. We subsequently report our experimental results on 2D image datasets in Section 3 and conclude in Section 4.

Related Work. This section presents a short introduction to the literature relevant to tackling the 3D assisted face recognition problem.
Wang *et al.* (Wang et al., 2014) present a detailed overview of facial landmarks localisation methods. Current landmark detection methods are either model-based (Cootes et al., 2001) or regression-based (Dollár et al., 2010) (Feng et al., 2015) (Feng et al., 2018). The model-based methods create a representation of the shape during training and use the shape to fit faces during testing. Popular frameworks include *3D Dense Face Alignment* (3DDFA) (Zhu et al., 2016) in which a dense 3D face model is fitted to the image with a CNN model, and *Pose-Invariant 3D Face Alignment* (Jourabloo and Liu, 2017) that estimates both 2D and 3D landmarks by integrating a 3D point distribution model. Zhang *et al.* (Zhang et al., 2014) used a cascade of several successive stacked auto-encoder networks that refines the coarse locations obtained from the first stacked auto-encoder network.

Bulat *et al.* (Bulat and Tzimiropoulos, 2017) first roughly rotated each set of facial landmarks and then refined the detection results. In 2018, Feng *et al.* (Feng et al., 2018) introduced a new loss function, namely Wing loss, for facial landmark localisation with CNNs. The landmark detectors used in this work for 2D images and for texture isomaps are based on the methods of Bulat *et al.*and Feng *et al.*.

Ding *et al.* (Ding and Tao, 2016) describe different pose-invariant face recognition (PIFR) algorithms and stated PIFR as crucial to realise the full potential of face recognition for real-world applications. They classify exsiting PIFR algorithms into four categories: pose-robust feature extraction, multi-view subspace learning, face synthesis and hybrid approaches.

Dataset augmentation techniques are transformations that are applied to images without changing the containing face's identity. Such methods are known to improve the performance of CNN-based methods and prevent over-fitting (Chatfield et al., 2014). Masi *et al.* (Masi et al., 2016) suggest a much more sophisticated technique to augment a generic face dataset. Their approach is to synthesise new face images, by creating face specific appearance variations in pose, shape and expression.

2 METHODOLOGY

The aim of this work is to investigate the merit of using 3D models and 2D reconstructed faces from 3D models for 2D face recognition. This requires a representation that recovers both 3D shape and texture information from a 2D image, as well as the ability to extract powerful features from this representation using CNNs.

The first step of the proposed pipeline in Figure 1 is to detect facial landmarks in an input image using the wing loss (Feng et al., 2018). We then fit a 3D Morphable Face model (3DMM) to the landmarked image using the algorithm and open-source software described in (Huber et al., 2016). As 3DMM, we opt for the Surrey Face Model, which is a compact PCA based representation of 3D face variability, learned from a set of 3D face scans, and consisting of separate shape and texture parts.

Every vertex in a 3D mesh stores a spatial coordinate (x,y,z) and a texture coordinate (u,v). The UV coordinates form a 2D embedding of the 3D vertex coordinates to store the texture in an image form. Such a generic representation of the face texture is created with an algorithm for rectangular texture maps that finds a projection from the 3D vertices to a 2D plane that preserves the geodesic distance between the mesh vertices. We follow the method for performing these steps described in (Kittler et al., 2018). As advocated in (Kittler et al., 2018), we use a conformal Laplacian Eigenmap where the boundary vertices are constrained to a square, as shown in Figure 2, calling it *square texture map*. The texture map contains the remapped texture of the original 2D image, preserving all details.

Although the accuracy of facial landmarking is very high with CNN-based landmark detectors, there are still some images with incorrect landmark points, producing variations in the quality of the resulting texture maps. The pose of a face also affects the quality of the remapped texture, at least in parts of the face. This paper focuses on the problem of texture map enhancement as a basis for improved 3D assisted 2D face recognition performance.

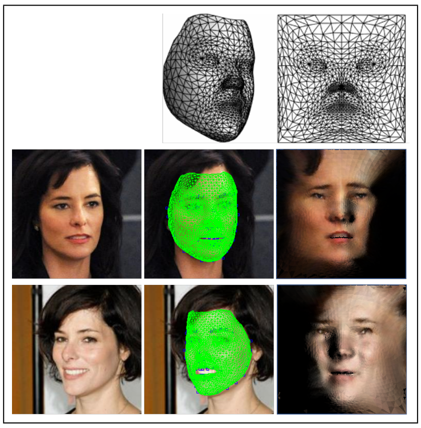

Figure 2: 3D face shape and constrained Laplacian square texture map.

2.1 Square Texture Map Quality Assessment

Although intuitively appealing in principle, a practical realisation of having surface texture is challenging because the 3D reconstruction from 2D projections is prone to errors. This leads to texture distortion, which is reflected in degraded recognition performance. We propose two skin texture enhancement methods to rectify the problem: a geometric correction to counteract the shape distortion, and texture merging to fill texture holes caused by pixel visibility issues in non-frontal poses. We propose a measure of texture quality and use it to control the texture fusion process.

To measure the quality of square texture maps we propose a number of different criteria:

Pose Rating. The visible area of a face depends on pitch, yaw and roll of the face from the camera perspective. A face that is directed to the camera gives us more texture information than a face that turns away from the camera. Also the probability of detecting better landmarks is higher on a frontal face. An image has n rows and m columns. Every pixel p_{ij} consists of red, green, blue and alpha values ranging from 0 to 255. The alpha value represents the angle to the camera for each pixel, by mapping angle of 0 to 90 degrees to values from 255 to 0. The pose rating value

P is the sum of all alpha values α_{ij} for all pixels, divided by the number of pixels and the maximum alpha value:

$$P = \frac{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \alpha_{ij}}{n \cdot m \cdot 255}. \tag{1}$$

Overall Histogram Rating. The histogram of the texture map tells a lot about the image quality. At first the image is transformed into a grey-scale image based on the RGB values from 0 to 255 per pixel. The histogram is a discrete function $h(r_k) = n_k$, where r_k is the kth grey level and n_k is the number of pixels in the image having grey level r_k (González and Woods, 2008). The next step is to find the maximum value max of n_k and calculate normalised values $p_k = n_k/max$. The values r_0 (black) and r_{255} (white) will not be counted to compute the rating H_{image} since these values represent the pixels of the face's invisible areas. The final rating is given by:

$$H_{image} = \frac{\sum_{k=1}^{254} p_k}{254}. \tag{2}$$

H_{image} represents a measure of the overall histogram rating of the distribution of tones. A wide distribution is an indicator for rich information content in the image. Peaks in the histogram are often detected in blurry or badly aligned texture maps.

Eye Position. We run an eye detector on the texture map in the expected area of the eyes. We define a score function that will be high if the detected position is exactly where an eye should be in the texture map, since the texture map is aligned. The higher the deviation of the eye position, the worse the rating. For the texture maps of a size 224 x 224 pixels we determined optimal coordinates for the left (p_{lx}, p_{ly}) and the right eye (p_{rx}, p_{ry}). Figure 3 shows some eye detection results inside the search area (red rectangle). The green rectangles stand for the detected eye positions. We take the centres of green rectangles as left and right eye positions l_x, l_y, r_x, r_y and calculate the rating E_{left} and E_{right}. The difference between the detected and the model eye coordinates is then divided by a factor N, that defines when a deviation measures reaches the value 0. We experimentally determined $N = 40$ to be a good value for an image size of 224x224.

$$E_{left} = 1 - \frac{|l_x - p_{lx}| + |l_y - p_{ly}|}{N} \tag{3}$$

$$E_{right} = 1 - \frac{|r_x - p_{rx}| + |r_y - p_{ry}|}{N} \tag{4}$$

Since there are images with eye occlusion like sunglasses and profile images with only one eye visible, the rating of an undetected eye will be excluded from the overall score.

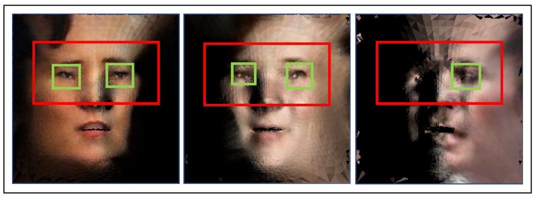

Figure 3: The green rectangles show the detected eyes found inside the red area.

Eye Detection Weight and Eye Histogram. E_{detect} is the value returned from the eye detector, for the probability of a detected eye, ranging from 0 to 1. Like the overall histogram rating, a histogram of the pixels in the area of the detected eyes will be analysed and rated as H_{eye}. If there are two eyes, the histogram will be combined, if there is no eye detected, the rating will be omitted and does not influence the overall score. Figure 4 shows the histogram charts for three different eyes, the score is likely to be higher, if the histogram values are evenly distributed and calculated by:

$$H_{eye} = \frac{\sum_{1}^{254} p_k}{254} \tag{5}$$

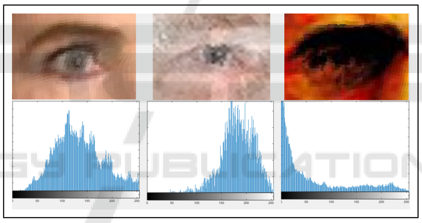

Figure 4: Eye histogram charts of different quality levels.

Overall Rating. The overall rating R is the average of all calculated values, which can consist of a maximum number of six ratings when both eyes have been detected. For the histogram and pose ratings the additional parameters have been determined during test runs with representative images.

2.2 Texture Map Reconstruction

To reconstruct self-occluded face parts and to improve the quality of the texture maps, we build a system for face texture improvement.

Deep Neural Network for Texture Map Landmarks. As a first step of the system, we train a CNN for landmark detection in texture maps. The network is build on the Face Alignment Network of Bulat *et al.* (Bulat and Tzimiropoulos, 2017), which uses a stack of four Hourglass networks combined with a

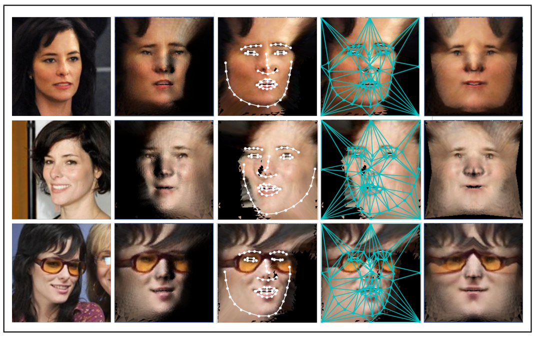

Figure 5: Texture Map Reconstruction. Left to right: CASIA WebFace image, input texture map, facial landmarks, Delaunay triangulation, enhanced output texture map.

hierarchical, parallel and multi-scale block. The network has been trained on about 5,000 manually annotated texture maps of the CASIA WebFace dataset.

Face Texture Alignment. The face texture alignment is based on a warping algorithm that uses facial landmarks and Delaunay triangulation. We use the 68 facial landmark points and eight points on the boundary of the texture map to calculate a Delaunay triangulation, which is used to break the texture into triangles. Having the 68 facial landmark coordinates of the defined texture map of the Surrey Face Model, combined with a standard Delaunay triangulation, these triangles can be used to calculate a 2×3 matrix M of an affine transformation for each triangle so that:

$$\begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = M \cdot \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \qquad (6)$$

where

$$dst(i) = (x'_i, y'_i), src(i) = (x_i, y_i), i = 0, 1, 2 \qquad (7)$$

Finally the source image src is transformed using the specified matrix M for each particular triangle to calculate the pixels of the destination image dst:

$$dst(x,y) = src(M_{11}x + M_{12}y + M_{13}, M_{21}x + M_{22}y + M_{23}) \qquad (8)$$

Texture Merging for Data Augmentation. Data augmentation is widely applied to training and test data to improve the performance of CNN-based methods and prevent over-fitting. The proposed approach to augmentation is based on merging textures of a person's face derived from different images of that person. In texture maps, all sides of a face are visible and eyes, nose, mouth are all aligned to a common reference frame, derived from the embedding. In the process of generating our square texture maps, every pixel gets its assigned RGB values, and an alpha value for visibility depending on the calculated angle to the camera.

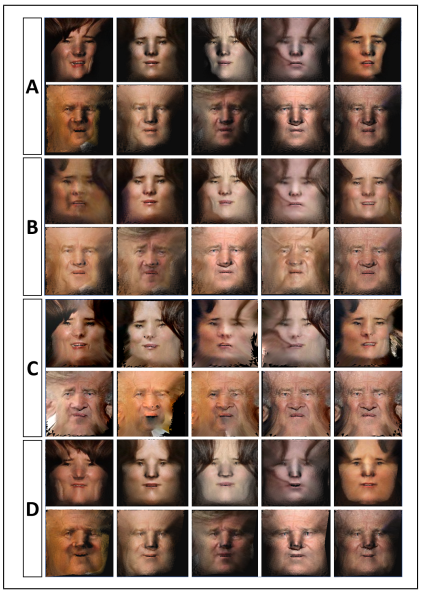

Figure 6: Data augmentation examples of one female and one male subjects with five images per subject. *A:* Generated square texture maps from a single 2D image. *B:* Merged four texture maps by alpha composite method. *C:* Merged two texture maps by Poisson blending. *D:* Landmark aligned and edited square texture map.

Our data augmentation aims to explore the effect of merging different numbers of textures from a set of images of the same person to a new one. Our first method to merge the texture maps is to compose the pixels based on their alpha values. The texture maps are merged by taking the weighted mean with the values from the alpha channels used as weights.

In the second approach, we used Poisson blending to merge textures. Poisson blending is a gradient domain image processing method that blends images by combining them in the gradient domain and solving for optimal RGB pixel values.

Our third training and evaluation dataset consists of the facial landmark aligned texture maps from our texture map reconstruction algorithm.

Figure 6 shows some example images from our augmented CASIA WebFace dataset.

3 EXPERIMENTS

The effect of our texture enhancement method can be judged subjectively from Figure 5. However, the key motivation for the development of these texture enhancement mechanisms is to improve the performance of 3D assisted 2D face recognition approaches. Consequently, as objective measures of the proposed square texture enhancements, we shall use the face recognition and face verification rates achieved by our 3D face recognition engine designed for use with these texture maps. In particular, the aim of our experiments is to evaluate the performance of a face recognition engine trained using texture maps created for the CASIA database. The training is limited to the CASIA dataset so that in terms of the information available for training, the results are comparable to the 2D face recognition methods in the literature, trained on the same resource. The evaluation is carried out on standard benchmarks, *i.e.* the LFW and IJB-A datasets, using the standard protocols.

Experimental Settings. The CASIA-WebFace (Yi et al., 2014) dataset is used for scientific research of unconstrained face recognition.

During each training, we conduct face verification tests on the Labeled Faces in the Wild (LFW) benchmark to evaluate the performance of the data and configuration. The LFW dataset contains more than 13,000 images of faces and has been the standard benchmark for unconstrained face verification for many years. The IARPA Janus Benchmark-A face challenge (IJB-A) was an open challenge in which researchers execute algorithms on NIST-provided image sets, and return output data to NIST for scoring. From 2015-2017 NIST produced regular results reports.

Face Recognition Engine. We use a TensorFlow implementation of a face recogniser, described in the paper from Schroff (Schroff et al., 2015), which is based on Inception-Resnet-V1. We train the model using softmax loss, which has been shown to be one of the best-performing loss in recent works (Szegedy et al., 2017). The input to the system are square texture maps with a resolution of 224 x 224 pixels as a training and test set for the face recognition network. We use a Tensorflow implementation of Inception-ResNet-v1.

Effect of Texture Map Reconstruction. In the first experiment, referred to as step (A), the aim was to establish a baseline and to gauge the effect of texture enhancement achieved by square texture reconstruction. For this reason we trained the system and measured its performance on individual square texture maps. The underlying aim of the experiment was to establish that facial recognition based on square-texture-maps is fundamentally possible and already achieves an evaluation rate which is not lagging too far behind 2D methods in the literature. We also wanted to measure the effect of square texture map reconstruction. For this reason we performed the experiment on the original individual square texture maps first, and then repeated it with the texture maps obtained using the reconstruction method described in Section 2.2. The verification on LFW was performed at a false acceptance rate (FAR) of 0.001 and the verification on IJB-A was performed at a false acceptance rate of 0.01. The results on the original texture maps achieved a verification rate of 87.7% and an accuracy of 97.7% on LFW and a verification rate of 73.3% on IJB-A as shown in Table 1.

Table 1: Evaluation results of training the square texture maps (A), the merged square texture maps (B+C), the landmark aligned reconstructed texture maps (D) and the combinations of them.

Method	LFW Acc.	IJB-A Ver.
A: square texture maps	97.7	73.3
B: 4-pic merged textures	96.8	68.7
C: 2-pic merged textures	97.0	69.5
D: landmark aligned textures	98.0	75.1
A+B	98.1	79.4
A+B+C	98.1	83.3
A+B+C+D	98.4	87.3

Effect of Texture Map Merging. In the second experiment, referred to as step (B), four textures were used to create a new texture by merging the faces based on their alpha values as described in Section 2.3. To make sure that the total number of the training images does not change, they are merged as follows: If there are five images of a person, the first four images are merged to a new image, the second to fifth image are merged to a second new image, image three, four, five and one then are merged to third new image, etc. The result of the evaluation was surprisingly worse than the results of A, but this can be explained by the fact that the evaluation was done on the non-merged textures, with missing values, which the network hasn't learned to interpret during training.

In order to adapt the evaluation data to the training data, we also merged the images of LFW and IJB-A according to the method described above. Where there were less than three images per class, additional images were generated by horizontal mirroring. Using these measures, we were able to improve the result of the first experiment on IJB-A verification from 73.3% in A to 75.2%.

Effect of Texture Map Augmentation. In the next step, by using the combination of A and B, we trained for the first time with data augmentation. Doubling the data has significantly improved all the test results.

For example the IJB-A verification result increased from 75.2% to 79.4%.

Then, in the next experiment (C), we merged two images by Poisson Blending and added them to the existing Set A and B. Thus, the data volume of the training images has tripled and the results on the IJB-A dataset have again improved from 79.4% to 82.1%. Table 1 shows the evaluation results obtained by training using the different variants of texture maps.

That is followed by experiment (D), where the original square texture maps are realigned by our facial landmark detector and Delaunay triangulation. This has lead to a significant improvement of the trained model, having an accuracy on LFW of 98.0 and 75.1 on IJB-A verification. Even the dataset augmentation benefits a lot of adding dataset (D) to achieve our best overall results.

Table 2: Face verification results on LFW.

Method	#images	ACC(%)
Hassner *et al.*(Hassner et al., 2015)		93.62
HPEN (Zhu et al., 2015)	0.75M	96.25
FF-GAN (Yin et al., 2017)	0.5M	96.42
CASIA-NET (Yi et al., 2014)	0.5M	96.42
DeepFace (Taigman et al., 2014)	4M	97.35
Masi *et al.* (Masi et al., 2016)	2.6M	98.06
VGG Face (Parkhi et al., 2015)	2.4M	98.95
Ours w/o augmentation	0.5M	97.7
Ours (A+B+C+D)	1.9M	98.4

Table 3: Evaluation on IJB-A dataset.

Method	IJB-A Ver.	IJB-A Id.
GOTS (Klare et al., 2015)	40.6	44.3
OpenBR (Klontz et al., 2013)	23.6	24.6
Wang *et al.* (Wang et al., 2015)	73.2	82.0
FF-GAN (Yin et al., 2017)	85.2	90.2
Masi *et al.* (Masi et al., 2016)	88.6	90.6
Ours	87.3	89.8

Finally, we compare our evaluation results to other experiments that used the same dataset for training and evaluation. Table 3 shows that we can compete with previous work. Using our augmentation strategies, we outperform most original methods, and achieve a performance close to Masi *et al.*, who additionally synthesised new poses, expressions and identities.

4 CONCLUSIONS

Although 3D assisted 2D face recognition has in theory the potential to surpass the performance of 2D face recognition by virtue of separating the key sources of face biometric information, namely face shape and skin texture, it has always lagged behind purely 2D techniques. There are several reasons for this state of affairs. First of all, recovering 3D information from 2D projections is prone to errors. Moreover, the conventional mesh representation of 3D faces is not convenient for convolutional processing by the latest machine learning tools, i.e. deep neural networks.

We proposed two enhancement techniques: geometric texture map rectification to correct for shape reconstruction errors, and quality controlled texture merging from multiple images. We showed that with these innovations the performance of our 3D face recognition engine, designed for, and working with, 3D face representations of 2D faces (texture only), can produce competitive results on standard benchmarking datasets. As there is a considerable scope for improving many aspects of the 3D assisted 2D face recognition approach, we consider these results as very promising. Future improvements will include training on much larger databases, following the path of purely 2D face recognition methods, as well as using the 3D shape information in conjunction with the skin texture maps.

ACKNOWLEDGEMENTS

This work was supported in part by the EPSRC Programme Grant (FACER2VM) EP/N007743/1, the EPSRC/dstl/MURI project EP/R018456/1, the National Natural Science Foundation of China (61902153) and the BMBF FHprofUnt program, no. 03FH049PX5.

REFERENCES

Bulat, A. and Tzimiropoulos, G. (2017). How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230, 000 3d facial landmarks). In *ICCV*, pages 1021–1030. IEEE Computer Society.

Chatfield, K., Simonyan, K., Vedaldi, A., and Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. In Valstar, M. F., French, A. P., and Pridmore, T. P., editors, *British Machine Vision Conference, BMVC 2014, Nottingham, UK, September 1-5, 2014*. BMVA Press.

Cootes, T. F., Edwards, G. J., and Taylor, C. J. (2001). Active appearance models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 23(6):681–685.

Ding, C. and Tao, D. (2016). A comprehensive survey on pose-invariant face recognition. *ACM Transactions on intelligent systems and technology (TIST)*, 7(3):37.

Dollár, P., Welinder, P., and Perona, P. (2010). Cascaded pose regression. In *Computer Vision and Pattern Recognition CVPR, 2010 IEEE Conference on*, pages 1078–1085. IEEE.

Feng, Z., Kittler, J., Awais, M., Huber, P., and Wu, X. (2018). Wing loss for robust facial landmark localisation with convolutional neural networks. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Salt Lake City, UT, USA, July 18-22, 2018*. IEEE Computer Society.

Feng, Z.-H., Huber, P., Kittler, J., Christmas, W., and Wu, X.-J. (2015). Random cascaded-regression copse for robust facial landmark detection. *IEEE Signal Processing Letters*, 22(1):76–80.

González, R. C. and Woods, R. E. (2008). *Digital image processing, 3rd Edition*. Pearson Education.

Hassner, T., Harel, S., Paz, E., and Enbar, R. (2015). Effective face frontalization in unconstrained images. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 4295–4304. IEEE Computer Society.

Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E. (2007). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst.

Huber, P., Hu, G., Tena, J. R., Mortazavian, P., Koppen, W. P., Christmas, W. J., Rätsch, M., and Kittler, J. (2016). A multiresolution 3d morphable face model and fitting framework. In *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2016) - Volume 4: VISAPP, Rome, Italy, February 27-29, 2016.*, pages 79–86. SciTePress.

Jourabloo, A. and Liu, X. (2017). Pose-invariant face alignment via cnn-based dense 3d model fitting. *International Journal of Computer Vision*, 124(2):187–203.

Kittler, J., Koppen, P., Kopp, P., Huber, P., and Rätsch, M. (2018). Conformal mapping of a 3d face representation onto a 2d image for CNN based face recognition. In *2018 International Conference on Biometrics, ICB 2018*, pages 124–131. IEEE.

Klare, B. F., Klein, B., Taborsky, E., Blanton, A., Cheney, J., Allen, K., Grother, P., Mah, A., Burge, M. J., and Jain, A. K. (2015). Pushing the frontiers of unconstrained face detection and recognition: IARPA janus benchmark A. In *CVPR*, pages 1931–1939. IEEE Computer Society.

Klontz, J. C., Klare, B., Klum, S., Jain, A. K., and Burge, M. J. (2013). Open source biometric recognition. In *IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems, BTAS 2013, Arlington, VA, USA, September 29 - October 2, 2013*, pages 1–8. IEEE.

Koppen, P., Feng, Z.-H., Kittler, J., Awais, M., Christmas, W., Wu, X.-J., and Yin, H.-F. (2018). Gaussian mixture 3d morphable face model. *Pattern Recognition*, 74:617–628.

Masi, I., Tran, A. T., Hassner, T., Leksut, J. T., and Medioni, G. G. (2016). Do we really need to collect millions of faces for effective face recognition? In *14th European Conference Computer Vision, ECCV 2016, Proceedings, Part V*, volume 9909 of *Lecture Notes in Computer Science*, pages 579–596. Springer.

Parkhi, O. M., Vedaldi, A., and Zisserman, A. (2015). Deep face recognition. In Xie, X., Jones, M. W., and Tam, G. K. L., editors, *Proceedings of the British Machine Vision Conference 2015, BMVC 2015*, pages 41.1–41.12. BMVA Press.

Schroff, F., Kalenichenko, D., and Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pages 815–823. IEEE Computer Society.

Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. In Singh, S. P. and Markovitch, S., editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 4278–4284. AAAI Press.

Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*, pages 1701–1708. IEEE Computer Society.

Wang, D., Otto, C., and Jain, A. K. (2015). Face search at scale: 80 million gallery. *CoRR*, abs/1507.07242.

Wang, N., Gao, X., Tao, D., and Li, X. (2014). Facial feature point detection: A comprehensive survey. *CoRR*, abs/1410.1037.

Yi, D., Lei, Z., Liao, S., and Li, S. Z. (2014). Learning face representation from scratch. *CoRR*, abs/1411.7923.

Yin, X., Yu, X., Sohn, K., Liu, X., and Chandraker, M. (2017). Towards large-pose face frontalization in the wild. In *ICCV*, pages 4010–4019. IEEE Computer Society.

Zhang, J., Shan, S., Kan, M., and Chen, X. (2014). Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment. In Fleet, D. J., Pajdla, T., Schiele, B., and Tuytelaars, T., editors, *Computer Vision - ECCV 2014 - 13th European Conference, Proceedings, Part II*, volume 8690 of *Lecture Notes in Computer Science*, pages 1–16. Springer.

Zhu, X., Lei, Z., Liu, X., Shi, H., and Li, S. Z. (2016). Face alignment across large poses: A 3d solution. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2016*, pages 146–155. IEEE Computer Society.

Zhu, X., Lei, Z., Yan, J., Yi, D., and Li, S. Z. (2015). High-fidelity pose and expression normalization for face recognition in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2015*, pages 787–796. IEEE Computer Society.