REAL-TIME 3D FACE RECONSTRUCTION FROM ONE SINGLE IMAGE BY DISPLACEMENT MAPPING

Tao Wu, Fei Zhou and Qingmin Liao*

Shenzhen Key Lab. of Information Sci&Tech
Department of Electronics Engineering, Graduate School at Shenzhen, Tsinghua University, China

ABSTRACT

In this paper, we present a fast and robust method to reconstruct a plausible three-dimension (3D) face from one single frontal face image. In training phase, we classify the faces into several groups based on the facial structures and propose to learn a mapping, known as the displacement mapping (DM) in this paper, for each group. DM relates two displacements: One displacements, denoted as 2D displacements, represent the differences between the positions of feature points on the 2D training faces and those on the reference 2D face that has been pre-defined for the corresponding group; another displacements, denoted as 3D displacements, are the differences between the positions of vertices on the reconstructed 3D face and those on the reference 3D face that is also predefined. During the reconstruction phase, we first classify the input face as one of the groups and calculate the 2D displacements. Then we take advantage of the 2D displacements and the learned DM to estimate the 3D displacements. Subsequently, 3D displacements can be used to obtain the precise 3D face by shifting the 3D reference face. Experiments on Basel face model (BFM) database as well as some real-world 2D face images demonstrate the effectiveness and efficiency of the proposed method, in comparison with some state-of-arts methods.

Index Terms— 3D face reconstruction, feature points, displacement mapping, reference model

1. INTRODUCTION

There-dimensional (3D) face reconstruction has always been an active topic because of its benefits to many real-world applications such as face recognition and virtual reality. However, it has always been a challenging and difficult task due to the complexity of facial geometric structure and diversity of individuals. Traditional 3D face reconstruction methods mainly focus on multiply views, such as [1], [2], [3]. These methods require face images from multiply views, and the registration of them is necessary. Therefore, the large costs with respect to the time and memory reduce the efficiency of these kinds of methods.

To avoid the registration problem, recently, researchers pay more attention to the techniques of 3D face reconstruction from single image, which can be categorized into three classes.

1) *Shape-from-shading*. These methods take advantage of the clues which are caused by the external conditions such as shading [4], [5], [6]. These approach utilises the idea that the depth information can be acquired through a given reflectance model. However, these methods are highly restricted by the input images since the illumination conditions are usually uncertain and the shooting angles are hard to determined.

2) *Example-based methods*: [7], [8], [9], [10], and [11]. These methods allow more realistic face reconstruction than other methods based on the development of 3D scanning technology which leads to the creation of more accurate 3D face models. However, the quality of the reconstruction results rely heavily on the chosen examples. For example, [6] and [12] considered that a generic 3D face model is hard to learn since an amount of 3D face models are needed. However, it is difficult to obtain enough 3D face samples.

3) *Learning-based method*. These methods attempt to establish mapping between 3D faces and 2D face images based on a training set of pairing 3D faces and 2D images. [13] and [14] learn the respective transformations for 3D faces and 2D face images in pairs to transform them into a common feature space. Then they construct a mapping between the 3D faces and their corresponding 2D face images. Off-line learning makes it more effectively when reconstructing a 3D face.

Inspired by the proposed methods, in this paper, we propose to relate 2D faces and 3D ones by displacement mapping (DM). Specifically, in training phase, we classify the training samples into several groups. For each group, we learn a DM which relates 2D and 3D displacements. In reconstruction phase, we classify the input face as one of the groups and calculate the 2D displacements which represent the spatial differences between the feature points on the input face and those on the reference 2D face. Then we calculate the 3D displacements that represent the spatial differences between the

This work was supported by the National Natural Science Foundation of China under Grant No.61271393 and 61301183 and the Special Foundation for the Development of Strategic Emerging Industries of Shenzhen under Grant No. JCYJ20140417115840272 and JCYJ20150331151358138

*Corresponding Author, E-mail: flying.zhou@163.com

vertices on the reconstructed 3D face and those on the reference 3D face. In comparison with the previous methods, we have the following advantages. First, we classify the faces into several groups and learn a series of DMs to relate 2D and 3D faces. The facial shapes are similar in each group and thus the displacements are small. Therefore, the mapping relations can be treated as linear. This seriously reduces not only the complexity of the model established by our method but also the complexity of our training phase. Secondly, our training phase is off-line, which makes our method can reconstruct a 3D face in real-time. Last but not the least, experiment results on BFM database and real-world images demonstrate the precision and robustness of the proposed method.

2. PROPOSED METHOD

In this section, the whole system of our method about 3D face reconstruction from a single frontal face image is presented. The following subsections will detail the method.

2.1. Clustering based on feature points

Suppose *w* 3D face models are used as our training samples, each of which has *v* vertices and *p* patches. For each 3D face model, we use the method of perspective projection to obtain a corresponding 2D frontal face image. Then the method in [15] is used to detect *r* facial feature points on each 2D face images automatically. The detection results of several images are shown in Fig.1. Subsequently, we randomly select *k* sets of feature points as initial clustering centers and then classify the *w* pairs of 2D and 3D faces into *k* groups by the algorithm of k-means based on the distribution of the feature points. For each group, the aim of training described in the following subsection is to obtain the mapping relationship between the 2D feature points and the corresponding 3D faces.

2.2. Training phase

In previous subsection, we have divided the training samples into several group. Since the training phases are the same for all groups, the following discussion only aims at one of the *k* groups. For one group, we suppose it has *m* pairs of 2D images and 3D faces. We denote the landmarks of the *i*th image as $\mathbf{F}_f^{(i)} = (f_{x_1}^{(i)}, f_{y_1}^{(i)}, ..., f_{x_r}^{(i)}, f_{y_r}^{(i)}), i = 1, 2, ..., m$, where f_{x_l} and f_{y_l} represent the abscissa and ordinate of the *l*th feature point respectively. Let $(f_{x_n}^{(i)}, f_{y_n}^{(i)})$ be the coordinates of the nose tip point, which is shown as the red point in Fig.2. We move it to original point and correspondingly adjust $\mathbf{F}^{(i)}$ as $\mathbf{F}_f^{(i)} = (f_{x_1}^{(i)} - f_{x_n}^{(i)}, f_{y_1}^{(i)} - f_{y_n}^{(i)}, ..., f_{x_r}^{(i)} - f_{x_n}^{(i)}, f_{y_r}^{(i)} - f_{y_n}^{(i)})^T$. Then we calculate the average feature points of the *m* 2D images as

$$\bar{\mathbf{F}} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{F}^{(i)} \tag{1}$$

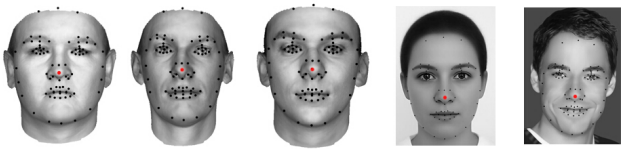

Fig. 1. Results of feature points extraction

where $\overline{\mathbf{F}}$= $(\bar{f}_{x_1}, \bar{f}_{y_1}, ..., \bar{f}_{x_r}, \bar{f}_{y_r})$. We record it as the 2D reference face model of the *q*th group. Then We calculate the differences between each $\mathbf{F}^{(i)}$ and $\overline{\mathbf{F}}$

$$\mathbf{X}^{(i)} = \mathbf{F}^{(i)} - \bar{\mathbf{F}}; i = 1, 2...m \tag{2}$$

where $\mathbf{X}^{(i)} = (x_1^{(i)}, y_1^{(i)}, ..., x_r^{(i)}, y_r^{(i)})^T$ is the 2D displacements of the training 2D faces.

Similarly, we denote the vertices of the *i*th 3D face model as $\mathbf{G}^{(i)} = (g_{x_1}^{(i)}, g_{y_1}^{(i)}, g_{z_1}^{(i)}, ..., g_{x_v}^{(i)}, g_{y_v}^{(i)}, g_{z_v}^{(i)})$, where g_{x_l}, g_{y_l}, and g_{z_l} represent the coordinates of the *l*th vertex in x-direction, y-direction and z-direction, respectively.

Let $(g_{x_n}^{(i)}, g_{y_n}^{(i)}, g_{z_n}^{(i)})$ be the vertex of nose tip corresponding to the feature point of nose tip on the 2D face. Similar to the 2D case, we also move it to original point and correspondingly adjust $\mathbf{G}^{(i)}$ as $\mathbf{G}^{(i)} = (g_{x_1}^{(i)} - g_{x_n}^{(i)}, g_{y_1}^{(i)} - g_{y_n}^{(i)}, g_{z_1}^{(i)} - g_{z_n}^{(i)}, ..., g_{x_v}^{(i)} - g_{x_n}^{(i)}, g_{y_v}^{(i)} - g_{y_n}^{(i)}, g_{z_v}^{(i)} - g_{z_n}^{(i)})^T$. Then we calculate the 3D displacements which represent the differences between each 3D face model and the average 3D face:

$$\mathbf{Y}^{(i)} = \mathbf{G}^{(i)} - \bar{\mathbf{G}}; i = 1, 2...m \tag{3}$$

where $\overline{\mathbf{G}}$ is the average shape of the *m* training 3D face samples:

$$\bar{\mathbf{G}} = \frac{\mathbf{1}}{\mathbf{m}}\sum_{\mathbf{i=1}}^{\mathbf{m}}\mathbf{G^{(i)}} \tag{4}$$

We record $\overline{\mathbf{G}}$ as the reference 3D face model of the *q*th group and denote $\mathbf{Y^{(i)}} = (x'^{(i)}_1, y'^{(i)}_1, z'^{(i)}_1, ..., x'^{(i)}_v, y'^{(i)}_v, z'^{(i)}_v)^T$.

Since the faces have similar facial structures in the same group, the 2D displacements and 3D displacements are small and thus the mapping can be treated as linear approximately for each group. Therefore, the training loss function is:

$$\arg\min_{\mathbf{M}} \sum_{i=1}^{m} \left\| \mathbf{Y}^{(i)} - \mathbf{X}^{(i)} \cdot \mathbf{M} \right\|_2^2 + \sum_{j=1}^{3v} \lambda_j \cdot \|\mathbf{M}_j\|_2^2 \tag{5}$$

where $\mathbf{M}$ is the DM for the *q*th group, $\mathbf{M}_j$ represents the *j*th column of $\mathbf{M}$, $\|\bullet\|_2$ is L_2 norm and λ_j is coefficient of regular term.

Through the above calculation, we have obtained 3D reference face $\overline{\mathbf{F}}$, 3D reference face $\overline{\mathbf{G}}$, and displacement mapping $\mathbf{M}$ for the given group. The same training process are preformed on the other groups. Since we have *k* groups, there are *k* 2D, 3D reference faces, and *k* DMs, denoted as $\overline{\mathbf{F}}^{<j>}$, $\overline{\mathbf{G}}^{<j>}$, $\mathbf{M}^{<j>}$, $j = 1, 2, ..., k$.

2.3. Reconstruction process

The reconstruction process can be seen as the inverse process of the training phase. When getting a frontal face image, we first detect *r* feature points using the method in [15] and classify the input face as one of the *k* groups. The extracted feature points are denoted as $\mathbf{I_f} = (\mathbf{I_{x_1}}, \mathbf{I_{y_1}}, ..., \mathbf{I_{x_r}}, \mathbf{I_{y_r}})^{\mathbf{T}}$. Let (I_{x_n}, I_{y_n}) be the coordinates of the nose tip point. We adjust $\mathbf{I}_f$ as $\mathbf{I} = (I_{x_1} - I_{x_n}, I_{y_1} - I_{y_n}, ..., I_{x_r} - I_{x_n}, I_{y_r} - I_{y_n})^T$. We denote $I'_{x_i} = I_{x_i} - I_{x_n}$ and $I'_{y_i} = I_{y_i} - I_{y_n}, i = 1, 2, ..., r$.

Then, we classify the input face into one of the groups, which have been clustered in the training phase, using the algorithm of nearest neighbour (NN). Suppose that the given input face belongs to the *q*th group. Afterwards, we need to calculate the scaling coefficient *s* to normalize $\mathbf{I}$ to the same scale as $\overline{\mathbf{F}}^{<q>}$.

s is computed as:

$$s = \frac{\sum_{i=1}^{r} \sqrt{[(\bar{f}_{x_i}^{<q>} - \bar{F}_x^{<q>})^2 + (\bar{f}_{y_i}^{<q>} - \bar{F}_y^{<q>})^2]}}{\sum_{i=1}^{r} \sqrt{[(I'_{x_i} - \bar{I}_x)^2 + (I'_{y_i} - \bar{I}_y)^2]}} \quad (6)$$

where $\bar{I}_x$ and $\bar{I}_y$ are average abscissa and ordinate of $\mathbf{I}$. $\bar{F}_x^{<q>}$ and $\bar{F}_y^{<q>}$ are average abscissa and ordinate of $\overline{\mathbf{F}}^{<q>}$. Using the obtained *s*, we can zoom $\mathbf{I}$ as $\mathbf{I'} = \mathbf{I} \cdot s$.

Then, we calculate the 2D displacements between $\mathbf{I'}$ and $\overline{\mathbf{F}}^{<q>}$:

$$\mathbf{X_{diff}} = \mathbf{I'} - \bar{\mathbf{F}}^{<\mathbf{q}>} \quad (7)$$

Finally the reconstructed 3D face Y_{rec} can be obtained:

$$\mathbf{Y_{rec}} = \bar{\mathbf{G}}^{<\mathbf{q}>} + \mathbf{X_{diff}} \cdot \mathbf{M}^{<\mathbf{q}>} \quad (8)$$

After reconstructing the 3D face model, the last step is to assigning a pixel value for each vertex. Here we project the 3D face onto X-Y plane and for each vertex, we set its RGB value using bilinear interpolation of the nearest pixels based on the input frontal face image.

The training and reconstruction processes of the faces belonged to the other groups are the same with above process and we will not detail here.

3. EXPERIMENTS

This section details the experiment setup and provides experimental results of the proposed method.

3.1. Experimental setup

In our work, we used the Basel Face Model (BFM) [16] to construct the 3D faces. Each of the constructed 3D faces has 53490 vertices. By using the BFM, we construct 500 synthetic 3D faces. 400 faces are used as the training samples and the rest are used for testing. Perspective projection is used to obtain the frontal face images from the 3D face models. The resolution of the projected 2D face images are 1200×900. In training phase, K-means is used to classify the training samples into *k* groups based on the distribution of feature points. *k* is set as 5 in this paper. In reconstruction phase, we use Nearest-neighbor (NN) to classify the input face as one of the k groups. All the experiments are taken on the laptop with 4G MEMORY and 2.0GHz CPU.

Table 1. RMSE of our method compared with some state-of-arts methods.

Method	MFF [8]	SSF [9]	S3DMM [17]	NMEA [13]	Our method
RMSE1	6.52	6.35	3.53	3.23	2.70
RMSE2	2.52	2.28	2.35	2.32	1.94
PDEM	4.82%	4.69%	2.51%	2.27%	1.84%

3.2. Experimental result

Our method is further compared with several state-of-art methods : Multi-Feature Fitting (MFF) [8], Sparse SIFT Flow Fitting (SSFF) [9], Non-linear Manifold Embedding and Alignment (NMEA) [13], and Simplified 3D Morphable Model(S3DMM) [17]. Here, we adopt the following two measure criteria: Rooted Mean Square Error (RMSE) and Per-pixel Depth Error Map (PDEM) which are defined as [17]. In this paper, we define two kinds of RMSE: one is the RMSE of the entire face, denoted as RMSE1; another is the RMSE of the facial components area, denoted as RMSE2. The comparison results on the 100 test faces are shown in Table.1. The illustration of the RMSE1, RMSE2 and PDEM of our method on several test faces are shown in Fig.3.

The visual results of our method in BFM dataset are shown in Fig.2. We also test our method on several real-world images which are downloaded from the Internet or taken by the authors. These results are shown in Fig.4. From the Figs, we can find that our method preserve not only the whole facial forms but also the facial details of the input images, which highlight the robustness and reliability of our algorithm.

Moreover, our method shows good performance in reconstruction speed. It takes only 0.04 seconds for converting a testing image into 3D face and achieves the real-time performance by our method. In [11], several seconds are needed to reconstruct a 3D face from one single image. In [17], it takes more than 10 seconds to reconstruct a 3D face from 2D input images. In [8] and [9], nearly one minute is needed to convert a 2D image into a 3D face.

4. CONCLUSIONS

In this paper, we present a method of real-time 3D face reconstruction from a single image. We take advantage of feature

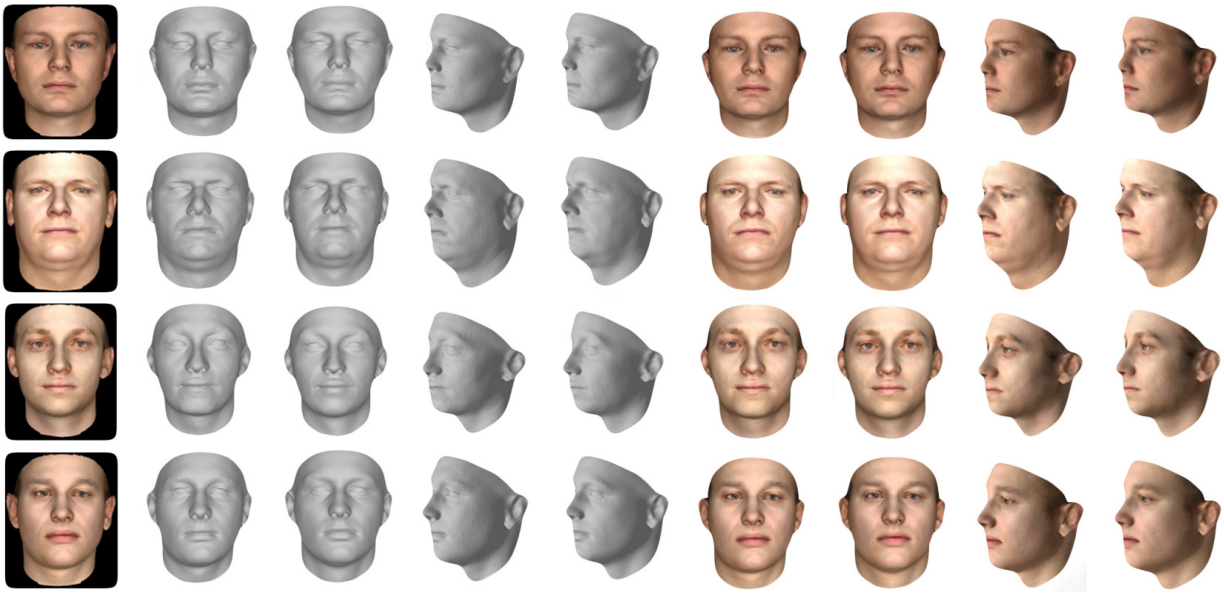

Fig. 2. 3D face reconstruction Result on BFM database. (a) Input images. (b) and (d) Ground-truth 3D face models in different views. (c) and (e) Reconstructed 3D face models. (f) and (h) Ground-truth 3D faces with texture in different views. (g) and (i) Reconstructed 3D faces with texture in different views.

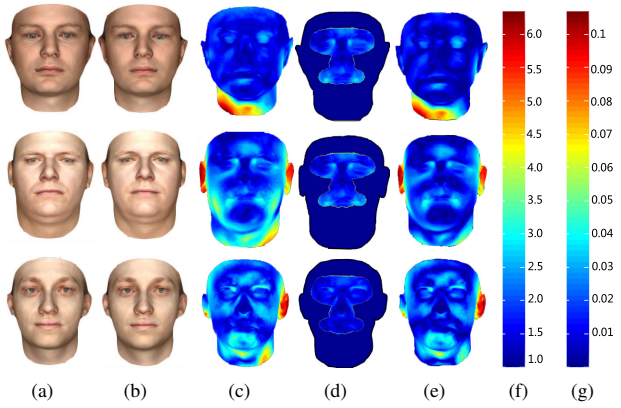

Fig. 3. RMSE1, RMSE2, and PDEM values of the proposed method. (a) Ground-truths. (b) Reconstruction Results. (c) RMSE1 values. (d) RMSE2 values. (e) PDEM values. (f) color bar of RMSE1 and RMSE2. (g) color bar of PDEM.

points detection algorithm and learning the mapping relationship between the displacement of the landmarks and that of the 3D face vertices. Our method needs only a single frontal face image as input and the entire process is executed in an automatic manner. The training process is off-line and the entire process is executed in an automatic manner. Our approach has been validated experimentally and shows good performance on both robustness, accuracy and speed.

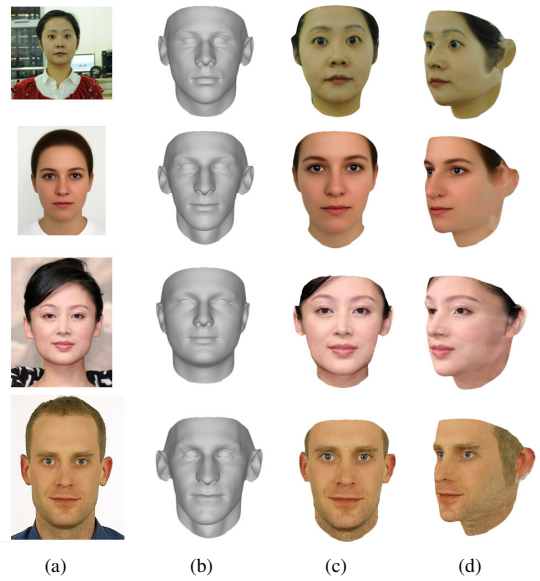

Fig. 4. Reconstruction results on real images. (a) Input images. (b) Reconstructed 3D face models. (c) and (d) Reconstructed 3D faces with texture in different views.

5. REFERENCES

[1] Yuping Lin, Gérard Medioni, and Jongmoo Choi, "Accurate 3d face reconstruction from weakly calibrated wide baseline images with profile contours," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 1490–1497.

[2] Zhengyou Zhang, Zicheng Liu, Dennis Adler, Michael F Cohen, Erik Hanson, and Ying Shan, "Robust and rapid generation of animated faces from video images: A model-based modeling approach," *International Journal of Computer Vision*, vol. 58, no. 2, pp. 93–119, 2004.

[3] Sung Joo Lee, Kang Ryoung Park, and Jaihie Kim, "A sfm-based 3d face reconstruction method robust to self-occlusion by using a shape conversion matrix," *Pattern Recognition*, vol. 44, no. 7, pp. 1470–1486, 2011.

[4] Berthold KP Horn, *Obtaining shape from shading information*, MIT press, 1989.

[5] Joseph J Atick, Paul A Griffin, and A Norman Redlich, "Statistical approach to shape from shading: Reconstruction of three-dimensional face surfaces from single two-dimensional images," *Neural computation*, vol. 8, no. 6, pp. 1321–1340, 1996.

[6] Ira Kemelmacher-Shlizerman and Ronen Basri, "3d face reconstruction from a single image using a single reference face shape," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, pp. 394–405, 2011.

[7] Volker Blanz and Thomas Vetter, "A morphable model for the synthesis of 3d faces," in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 1999, pp. 187–194.

[8] Sami Romdhani and Thomas Vetter, "Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. IEEE, 2005, vol. 2, pp. 986–993.

[9] Xiangyu Zhu, Dong Yi, Zhen Lei, Stan Z Li, et al., "Robust 3d morphable model fitting by sparse sift flow.," in *ICPR*, 2014, pp. 4044–4049.

[10] Unsang Park, Yiying Tong, and Anil K Jain, "Age-invariant face recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 5, pp. 947–954, 2010.

[11] Tal Hassner, "Viewing real-world faces in 3d," in *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2013, pp. 3607–3614.

[12] Jose Gonzalez-Mora, Fernando De La Torre, Nicolas Guil, and Emilio L. Zapata, "Learning a generic 3d face model from 2d image databases using incremental structure-from-motion," *Image and Vision Computing*, vol. 28, no. 7, pp. 1117–1129, 2010.

[13] Xianwang Wang and Ruigang Yang, "Learning 3d shape from a single facial image via non-linear manifold embedding and alignment," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 414–421.

[14] Mingli Song, Dacheng Tao, Xiaoqin Huang, Chun Chen, and Jiajun Bu, "Three-dimensional face reconstruction from a single image by a coupled rbf network," *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2887–2897, 2012.

[15] S. Milborrow and F. Nicolls, "Active Shape Models with SIFT Descriptors and MARS," *VISAPP*, 2014.

[16] Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter, "A 3d face model for pose and illumination invariant face recognition," in *Advanced video and signal based surveillance, 2009. AVSS'09. Sixth IEEE International Conference on*. IEEE, 2009, pp. 296–301.

[17] Jaeik Jo, Heeseung Choi, Ig-Jae Kim, and Jaihie Kim, "Single-view-based 3d facial reconstruction method robust against pose variations," *Pattern Recognition*, vol. 48, no. 1, pp. 73–85, 2015.