LEARNING-BASED FULLY 3D FACE RECONSTRUCTION FROM A SINGLE IMAGE

Xiaoping Hu, Ying Wang, Feiyun Zhu and Chunhong Pan

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences
{xiaoping.hu, ywang, fyzhu and chpan}@nlpr.ia.ac.cn

ABSTRACT

This paper presents an algorithm for fully reconstructing a 3D face from a single image. This task is still highly challenging as most current methods only care about the frontal face, ignoring side face, such as the neck, ears etc. In our algorithm, to get the more detailed texture, we deal with the shape reconstruction and texture recovery respectively. For shape, we estimate the deformation of the 3D model by a set of feature points. For texture, due to the similar facial structure, we divide the full texture into patches and show how sparse learning model can be used to fully recover the texture of the 3D face. Extensive experiment results on the CMU-PIE database and images downloaded from the Internet demonstrate that our method outperforms the state-of-the-art methods.

Index Terms— 3D Morphable model, face alignment,deform transfer, full reconstruction, sparse learning

1. INTRODUCTION

The 3D face reconstruction is a powerful tool for a wide range of computer vision tasks, such as 3D face recognition, 3D face tracking, medical plastic and so on[1, 2, 3, 4]. A number of methods have been proposed to acquire 3D face [3], despite we can obtain high resolution face models with 3D sensors, reconstructing the shapes and textures from 2D images is still a problem worthy of research.

Reconstructing 3D face from 2D images is an extremely ill-posed problem. Most recent methods suppose the existing 3D face model as the priority guiding reconstruction. For example, Kemelmacher-Shlizerman [5] proposed a reconstruction framework based on shape-from-shading (SFS). She utilized the input image as a guide to "mold" a single reference model to reach a reconstruction of 3D shape. In the SFS framework, it essentially needs the frontal face in the input image to avoid the reconstruction failure. However, this strong requirement upon the input image prevents SFS from providing a good reconstruction of the profile face.

Blanz and Vetter proposed a very powerful framework for face reconstruction with only one input image [4, 6]. They trained a 3D face database to build a morphable model, and optimized a cost function between the input image and morphable model. Experiments demonstrated that the 3DMM framework did reconstruct the full 3D face robustly. However, 3DMM with Stochastic Newton Optimization (SNO) is time-consuming and easy to trap into local minima, which makes faces appeared unrealistic.

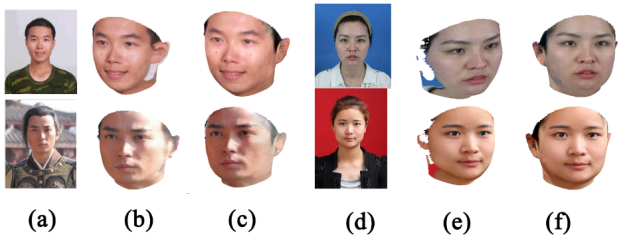

Fig. 1. The (a) and (d) columns are images downloaded from Internet with different illumination. (b) and (e) columns are the result of texture interpolation. Fully reconstruction for 3D face with our algorithm are presented by the (c) and (f).

There have been some attempts to address local minima problem. Blanz et al. [7] focused on improving the accuracy and efficiency of the fitting process respectively. In this case, they avoided the problems of local minima by using features derived from the input images rather than intensity data itself. But the system with a manually intensive procedure is far from flexible, since the user needs to manually specify point matching across multiple images and 2D-3D feature correspondences.

For the texture recovery, almost all methods for 3D face reconstruction only care about the frontal and visible texture , but not the neck, ears and other missing parts, which is equally important for reconstruction, recognition and so on. In [8], Jiang employed a linear interpolation algorithm to recover the missing areas by using neighborhood valid texture. But experiments demonstrate that interpolation doesn't work well. Because in most cases, the missing holes are to vast to fix by the neighborhood vertexes.

In this paper, we propose an efficient framework (Fig. 2) for reconstructing the shape and texture of 3D face automatically and fully. In our framework, we deform the model through a set of sparse feature points, obtained by the reliable

This work was supported by the National Natural Science Foundation of China under Grants 61272049, 61370039, and 61175025.

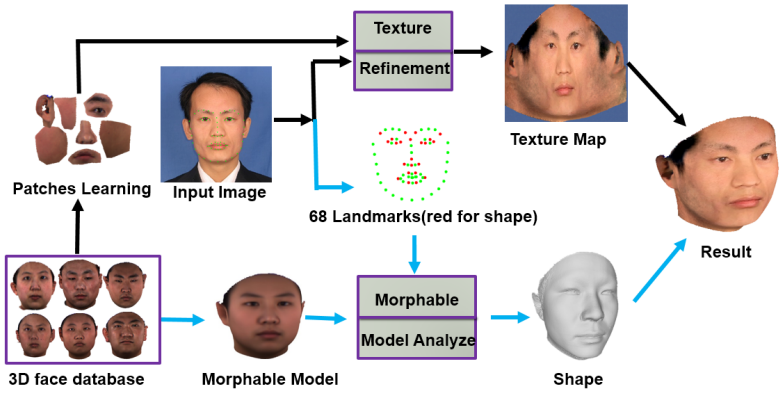

Fig. 2. Framework of our method, which consists of 5 steps as follows: 1) train a morphable model on the 3D database; 2) detect 68 landmarks on the face in the input image; 3) do an analysis-by-synthesis loop to optimize shape through 23 landmarks; 4) extract and infer texture from input image; 5) refine the full 3D reconstruction.

landmarks detection algorithm [9]. We find that the position of landmarks on eye, nose and mouth are almost position invariant, but not the ones on the side face. Thus we evaluate the relationship between reconstruction accuracy and different number of points and employ the most appropriate selection for better performance.

For reconstructing the texture, we use the linear combination to infer the shaded texture. As the facial structures among all human beings are almost the same [10] which means that inferring the missing texture based on learning strategy is a reasonable choice. Firstly, we train the model with a well-registered texture in 3D face database, and reconstruct missing texture via three steps: (1) extracting valid texture from the input image; (2) applying the learned model to infer the missing texture roughly; (3) computing the gradient field to refine the whole texture (Fig. 3).

The rest of this paper is organized as follows: In Section 2, we presents an automatic, non-iterative algorithm for shape reconstruction with a set of feature points. The learning-based texture recovery algorithm is proposed in section 3. Finally, the experiments for images in the CMU-PIE and downloaded from the Internet are presented in Section 4.

2. MODEL-BASED SHAPE RECONSTRUCTION

2.1. 3D Morphable Model

The morphable model is based on a data set of 3D face scans. In this paper, we apply the BJUT database [11] as the training set. We represent each 3D face with a shape vector and a texture vector. The shape vector consists of the coordinate value in the 3D world $(x_i, y_i, z_i) \in \mathbb{R}^3$. Similarly, the texture vector consists of every vertice's color values $(r_i, g_i, b_i) \in \mathbb{R}^3$. Now, 3D face model can be expressed as:

$$\mathbf{s} = \{(x_1, y_1, z_1), \cdots, (x_n, y_n, z_n)\},$$
$$\mathbf{t} = \{(r_1, g_1, b_1), \cdots, (r_n, g_n, b_n)\}.$$

Through above representation, we transform our 3D face into shape space and vector space. After point-to-point 3D correspondence, we can obtain the new face by a linear combination of the shapes and textures as follows:

$$\mathbf{s} = \sum_{i=1}^{m} a_i \mathbf{s}_i, \quad \mathbf{t} = \sum_{i=1}^{m} b_i \mathbf{t}_i. \tag{1}$$

Varying the coefficients $\mathbf{a} = (a_1, a_2, \cdots, a_m)^T \in \mathbb{R}^m$ and $\mathbf{b} = (b_1, b_2, \cdots, b_m)^T \in \mathbb{R}^m$ can generate arbitrary new faces. We perform a Principal Component Analysis (PCA) on the database of shape and texture vectors separately. Finally, we represent the model face with the average face and the combination of eigenvectors:

$$\mathbf{s} = \bar{\mathbf{s}} + \sum_{i=1}^{m-1} \alpha_i \mathbf{s}_i, \quad \mathbf{t} = \bar{\mathbf{t}} + \sum_{i=1}^{m-1} \beta_i \mathbf{t}_i, \tag{2}$$

where $\alpha, \beta \subseteq \mathbb{R}^{m-1}$ are the coefficient vectors.

2.2. Reconstruction with a Set of Sparse Points

Blanz [7] presents an approach for shape recovery under arbitrarily pose by a given set of 2D points on the input image. Our method is inspired by [7], but combines the state-of-the-art face alignment techniques to make process automatical. We fit 68 landmarks on each input image with the algorithm in [9] and apply 23 landmarks during deform the shape model. We compute the root mean square error (RMSE) of the 3D face reconstructed from 68 and 23 landmarks (red points in Fig. 2) with ground truth. Table 1 illustrates that the better choice is 23 points. Because the points around eyes, mouth are view invariant, making the algorithm much robust.

For each feature point, we can find its correspondent vertex on the model by projecting 3D vertexes onto 2D face and applying the same landmarks detection algorithm. The calculation of Mahalanobis distance between the feature points on

Table 1. The accuracy with different number of points.

the number of landmarks	68 points	23 points
average RMSE to ground truth	3.73	3.61

input image and the 3D model turn to be a linear combination problem. So, we can solve it by a least square minimization.

Set L as the orthographic projection in this paper, the sparse 3D model is $\mathbf{r} = \mathbf{L}\mathbf{s}_i, \mathbf{L} : \mathbb{R}^n \longmapsto \mathbb{R}^l$. After obtaining the landmarks both on input image and average face, we can calculate $\mathbf{y}$ as:

$$\mathbf{y} = \mathbf{r} - \mathbf{L}\bar{\mathbf{s}} = \mathbf{L}\mathbf{x}. \tag{3}$$

This is a overdetermined equation, which can be solved by least square regression. As detained descried in [7], the final cost function is:

$$\mathbf{E} = \parallel \mathbf{Q}\alpha - \mathbf{y} \parallel^2 + \lambda \parallel \alpha \parallel^2 + const. \tag{4}$$

Here, we treat the λ as the weight factor. $\mathbf{Q}$ is the sparse model matrix, and α is the correspondent coefficient. Using a Singular Value Decomposition $\mathbf{Q} = \mathbf{U}\mathbf{W}\mathbf{V}^T$ with a diagonal matrix $\mathbf{W} = \text{diag}(w_i)$, we can solve the coefficient $\mathbf{c}$ in a single step that:

$$\alpha = \mathbf{V} diag(\frac{w_i}{w_i^2 + \lambda})\mathbf{U}^T\mathbf{y}. \tag{5}$$

We deal with the unknown translation, rotation and scale parameters by treating them as additive terms in the eigenvectors of average model. Finally, we recover the shape with Eq. (2). This is a direct, non-iterative algorithm that makes the reconstruction much efficiently [7].

2.3. Fitting Process

We fit the 3D model to the 2D image in a analysis-by-synthesis loop. With 23 landmarks, the algorithm can automatically reconstruct 3D face in high resolution. The estimation for the rotation in first time of computation is under the assumption of small angels γ, θ, ϕ, which is unprecise for large rotation [7]. Therefore, we do iterations twice for full shape recovery. After fitting process, we can not only get the linear coefficients for shape, but also the pose of the image and the focal length of the camera.

3. LEARNING-BASED TEXTURE RECOVERY

Because of occlusion, the 3D face texture is inevitably incomplete reconstructing from a single image. A learning-based algorithm is presented here to learn a transform model to infer missing texture from the frontal reliable ones.

Model definition: Each full texture in the database is divided into N_s patches. The patch in the same position with different subjects lie in a similar facial structure space, which means any new RGB patch can be approximately spanned by a set of template patches under the assumption of diffuse-only

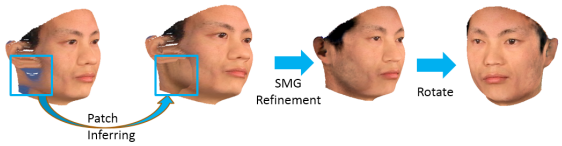

Fig. 3. The process for texture inferring and refinement.

reflectance. Let $\mathbf{p}_j$ represent the jth valid patch extracted from the visible face. We determine one patch whether valid or not through the result of landmarks detection. Thus, we can obtain the ith missing patch of texture with

$$\mathbf{y}_i = f_1\mathbf{p}_1 + f_2\mathbf{p}_2 + \cdots + f_N\mathbf{p}_N + \varepsilon = \mathbf{B}\mathbf{F} + \varepsilon, \tag{6}$$

where $\mathbf{B} = (\mathbf{p_1} ... \mathbf{p_N}) \subseteq \mathbb{R}^{3d \times N}$, $\mathbf{p_j} \subseteq \mathbb{R}^{3d}$ is an 1D formed by RGB values, d is the number of vertex that one patch contains. $\mathbf{F} = (f_1, f_2, ...f_N)^T$ is the coefficient vector, and ε is a noise term representing the specular reflection on some faces. For the complicated light condition in reality, ε is always nonzero entries. To explicitly capture the illumination constraints, we adopt the technique of trivial templates [12] here, such that each trivial template has only one nonzero element. The trivial templates $\mathbf{I} = (\mathbf{I}_1, \mathbf{I}_2, ..., \mathbf{I}_d) \subseteq \mathbb{R}^{3d \times 3d}$ are augmented into $\mathbf{B}$ as follows:

$$\mathbf{y}_i = (\mathbf{B}, \mathbf{I}, -\mathbf{I}) \begin{pmatrix} \mathbf{F} \\ \mathbf{e}^+ \\ \mathbf{e}^- \end{pmatrix} = \mathbf{C}\eta, \qquad s.t. \quad \eta \geq 0, \tag{7}$$

where $\mathbf{e}^+ \subseteq \mathbb{R}^{3d}$,$\mathbf{e}^- \subseteq \mathbb{R}^{3d}$ are called a positive trivial coefficient vector and a negative trivial coefficient vector respectively. $\mathbf{C} = (\mathbf{B}, \mathbf{I}, -\mathbf{I}) \subseteq \mathbb{R}^{3d \times (N+6d)}$ and $\eta = (\mathbf{F}, \mathbf{e}^+, \mathbf{e}^-)^T \subseteq \mathbb{R}^{(N+6d)}$ is a non-negative coefficient vector. The argument for enforcing nonnegativity constraints on η comes from their ability to deal with complicated light condition more than diffuse-only.

The system in Eq. (7) is underdetermined. We intend to choose the most similar templates from $\mathbf{B}$ for matching $\mathbf{y}_i$. So the coefficient η should be as sparse as possible. The error caused by environment illumination typically corrupts a fraction of the patches, there are a limited number of nonzero coefficients in $\mathbf{e}^+$ and $\mathbf{e}^-$ that account for the noise patch. Consequently, we rewrite the sparse model as:

$$\min_{\eta} \parallel \mathbf{y}_i - C\eta \parallel_2^2 + \lambda ||\eta||_1, \tag{8}$$

where $\parallel \eta \parallel_1$ and $\parallel \mathbf{y}_i - C\eta \parallel_2$ denote the ℓ_1 and ℓ_2 norms.

Solution: The solution of above problem is a intractable challenge, here, we choose the Augmented Lagrangian Method (ALM) [13, 14] to solve (8), resulting in several easily tackled unconstrained subproblems. We add an auxiliary variable as $\tau = \eta$. Thus, the Eq. (8) becomes:

$$\min_{\eta, \tau} \frac{1}{2} \|\mathbf{y}_i - \mathbf{C}\tau\|_2^2 + \lambda \|\eta\|_1 + \frac{\mu}{2} \parallel \tau - \eta \parallel^2, \quad \text{s.t. } \tau = \eta, \tag{9}$$

Fig. 4. The face results with the PIE-CMU database.

where μ is a penalty parameter. To guarantee the equality constraint, it requires μ approaching infinity, which may cause bad numerical conditions. Fortunately, there is no need to require $\mu \longrightarrow \infty$ when we introduce a Lagrangian multiplier instead. We get the standard ALM equation as:

$$\mathcal{L}(\tau, \eta, \Lambda, \mu) = \frac{1}{2} \|\mathbf{y}_i - \mathbf{C}\tau\|_2^2 + \lambda \|\eta\|_1 + \frac{\mu}{2} \left\| \tau - \eta + \frac{\Lambda}{\mu} \right\|_2^2, \quad (10)$$

where μ is a penalty parameter, $\mathbf{\Lambda}$ is Lagrangian multiplier. We will solve the η and μ separately. The algorithm flow is introduced in the Algorithm 1.

Algorithm 1 Learning-based Texture Recovery

Input: Template patches of texture $\mathbf{B} = [\mathbf{p}_1, \ldots, \mathbf{p}_N] \subseteq \mathbb{R}^{3d \times N}$, and the i^{th} missing patch.

Init variables $\mathbf{\Lambda_0}, \mu_0, \eta_0, \tau_0, \mathbf{C} = (\mathbf{B}, \mathbf{I}, -\mathbf{I})$.
do
  solver for $\eta : \mathbf{H} \leftarrow \tau + \frac{\mathbf{\Lambda}}{\mu}$,
  get the equivalence problem: $f(\eta) = |\eta| + \frac{\lambda}{2}(h - \eta)$;
  $\eta^* = \max\left(|\mathbf{h}| - \frac{1}{\gamma}, 0\right) \cdot \text{sign}(\mathbf{h})$;
  solver for $\tau : \mathbf{P} \leftarrow \eta - \frac{\mathbf{\Lambda}}{\mu}, \mathbf{y} \leftarrow \mathbf{C}\eta$;
  $\tau = \left(\mathbf{y}\mathbf{C}^T + \mu\mathbf{P}\right)\left(\mathbf{C}\mathbf{C}^T + \mu\mathbf{I}\right)^{-1}$;
  update $\Lambda : \mu \leftarrow k\mu(k > 1)$;
  $\mathbf{\Lambda} \leftarrow \mathbf{\Lambda} + \mu(\tau - \eta)$;
while$(\mathbf{abs}(\tau - \eta) < \varepsilon)$

Output: The refinement texture $\hat{\mathbf{y}} = \mathbf{C} * \eta$.

After obtaining the inferring textures, we combine them with the valid ones directly. Therefore, gaps exist between patches, which makes the texture appeared unrealistic. Then, we use streaming multi-grid for gradient-domain operation (SMG) to solve the Poisson equation and smooth the full texture which is detailed described in [15].

4. EXPERIMENTS AND RESULTS

We use the BJUT-3D as the face database. It contains 250 males and 250 females [11], with the age ranging from 16 to 49. All faces are in the natural expression and under the natural light. Approximately, each face consists of 65,000 vertices and 130,000 triangles. Besides, we have the CMU-PIE 2D faces database [16] and images downloaded from the Internet for testing.

For quantitatively analysis, given a reconstructed shape and its ground truth, we compute the root mean square error (RMSE) of each vertex as the errors of fitted shapes. The RMSE distance are normalized by the eye-to-eye distance. We select 100 faces which do not appear in the training set as the ground truth. The results in Table 3 indicates the stable performance for reconstructing from different views.

Table 2. Average rating of 200 reconstruction examples.

	very good	good	acceptable	bad
%	25.00	34.50	26.00	14.50

Table 3. Average error over all features in different views .

view	ba	bb	bc	bd	be
angle	$1.1°$	$38.9°$	$27.4°$	$18.9°$	$11.2°$
error	3.34	4.05	3.73	3.58	3.55
view	bf	bg	bh	bi	be
angle	$7.1°$	$-16.3°$	$-26.5°$	$-37.9°$	$0.1°$
error	3.61	3.71	3.79	3.98	3.44

For qualitative analysis, because of the full texture refinement process, our system produces plausible and photo realistic 3D models consistent with the input images (Fig. 1 and Fig. 4). We test our method on images downloaded from the Internet and PIE database from CMU [16] which vary in pose and illumination. We divide the results into four groups: very good, good, acceptable and bad. The average rating of 200 examples is showed in Table 2, which illustrates that most images obtain good result with texture. However, still part of results are bad because of the strong light noise and the wide rotation with yaw (more than 45 degree).

For the runtime, the landmarks detection with ESR takes less than 20 ms per-image (300 $\times$ 300 pixels) on a standard PC. And the process for shape fitting, texture extracting and inferring to full reconstruction takes around 20 seconds, while SNO took about 4 minutes per image in 3DMM framework.

5. CONCLUSION

We proposed a model-based shape reconstruction and learning-based full texture recovery framework. We use 23 feature points to reconstruct the high resolution 3D shape automatically, avoiding the local minimum. Based on sparse learning, we can fully reconstruct face's texture, producing plausible and photo realistic 3D models, solving the problem few researchers focus on. Both qualitative and quantitative experiments show that our method is able to produce high-quality 3D face models. We have tried to transplant this application to the mobile device. We firmly believe that this 3D face reconstruction framework can be applied to face recognition, medical beauty and so on.

6. REFERENCES

[1] Georgia Sandbach, Stefanos Zafeiriou, Maja Pantic, and Lijun Yin, "Static and dynamic 3d facial expression recognition: A comprehensive survey," *Image Vision Comput.*, vol. 30, no. 10, pp. 683–697, 2012.

[2] Chauã C. Queirolo, Luciano Silva, Olga Regina Pereira Bellon, and Mauricio Pamplona Segundo, "3d face recognition using simulated annealing and the surface interpenetration measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 2, pp. 206–219, 2010.

[3] Graham Fyffe, Andrew Jones, Oleg Alexander, Ryosuke Ichikari, and Paul E. Debevec, "Driving high-resolution facial scans with video performance capture," *ACM Trans. Graph.*, vol. 34, no. 1, pp. 8:1–8:14, 2014.

[4] Volker Blanz and Thomas Vetter, "Face recognition based on fitting a 3d morphable model," *IEEE Trans.Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1063–1074, 2003.

[5] Ira Kemelmacher-Shlizerman and Ronen Basri, "3d face reconstruction from a single image using a single reference face shape," *IEEE Trans.Pattern Anal. Mach. Intell*, vol. 33, no. 2, pp. 394–405, 2011.

[6] Volker Blanz and Thomas Vetter, "A morphable model for the synthesis of 3d faces," *Proc. Int'l Conf. SIGGRAPH'99*, pp. 187–194, 1999.

[7] Volker Blanz, Albert Mehl, Thomas Vetter, and Hans peter Seidel, "A statistical method for robust 3d surface reconstruction from sparse data," *Proc. Int'l Conf. 3DPVT*, pp. 293–300, 2004.

[8] Dalong Jiang, Yuxiao Hu, Shuicheng Yan, Lei Zhang, Hongjiang Zhang, and Wen Gao, "Efficient 3d reconstruction for face recognition," *Pattern Recogn*, vol. 38, no. 6, pp. 787–798, 2009.

[9] Xudong Cao, Yichen Wei, Fang Wen, and Jian Sun, "Face alignment by explicit shape regression," *Proc. Int'l Conf. IJCV*, pp. 177–190, 2014.

[10] Curzio Basso, Thomas Vetter, and Volker Blanz, "Regularized 3d morphable models," *Higher-Level Knowledge in 3D Modeling and Motion Analysis (Workshop)*, pp. 3–10, 2003.

[11] Yin Baocai, Sun Yanfeng, Wang Chengzhang, and Ge Yun, "BJUT large scale 3D face database and information processing," *Journal of Computer Research and Development*, vol. 46, no. 6, pp. 1009–1018, 2009.

[12] Xue Mei and Haibin Ling, "Robust visual tracking using l1 minimization," *Proc. IEEE Intl Conf. ICCV*, pp. 1436–1443, 2009.

[13] Nocedal Jorge and Wright Stephen, *Numerical Optimization*, 2006, New York:Springer, 2nd edition.

[14] Feiyun Zhu, Bin Fan, Xinliang Zhu, Ying Wang, Shiming Xiang, and Chunhong Pan, "10, 000+ times accelerated robust subset selection," in *Proc. Intl Conf. AAAI*, 2015, pp. 3217–3224.

[15] Michael Kazhdan and Hugues Hoppe, "Streaming multigrid for gradient-domain operations on large images," in *ACM Trans. on Graphics*. ACM, 2008, vol. 27, pp. 1–10.

[16] Terence Sim, Simon Baker, and Maan Bsat, "The CMU pose, illumination, and expression (PIE) database," in *Proc. Int'l Conf. AFGR*, 2002, pp. 53–58.