2D + 3D FACE MORPHING

[1] Yen-Cheng Liu (劉彥成) [2] Pei-Hwai Ciou (邱沛淮) Chiou-Shann Fuh (傅楸善)

[1, 2] Graduate Institute of Electrical Engineering
National Taiwan University
Taipei, Taiwan
r04921003@ntu.edu.tw r04921008@ntu.edu.tw fuh@csie.ntu.edu.tw

ABSTRACT

Face morphing is one of popular technique in camera-based mobile application. While both 2D and 3D morphing algorithm are well-developed in face morphing, existing algorithm still have some constraints. For instance, 3D model depends on multiple photos or scans to reconstruct. 2D model fail to represent head skeleton shape. As a result, we propose to take advantage of information from 2D and 3D model. We first detect multiple face in single image. Then, facial images will be merged under 2D and 3D algorithm. In the end, 2D and 3D face model will be combined into single result under this framework. We show our result in the experiment, and it correctly preserve skin appearance and head skeleton shape from multiple persons in single image. This can be applied to animation, child appearance prediction, and social media gaming.

Keywords – Face morphing, 2D Morphing, 3D Morphable Model

1. INTRODUCTION

Existing morphing algorithm are based on either 2D or 3D model. In general, 3D model has more parameters to represent facial feature compared to 2D model. The representation power of 3D model, in other word, is more plentiful than 2D model. Moreover, 3D model contains normal vector, which is natural to extract the head rotation and represent the head pose variation [3]. However, 3D model fails to represent skin appearance which is important component for face morphing. As a result, we try to take advantage of their complementary information and propose a synthesized 2D+3D morphing model.

In the proposal, we divide the work into three parts. First, face detection produces multiple small bounding boxes which will be stored individually. Second, two or multiple face patterns will be selected and converted to 2D and 3D morphable model respectively. In the end, we will fuse 2D and 3D models into single result which preserve respective facial feature and adequately fuse both facial features which includes skin appearance and head skeleton.

Detecting face in single image is the first step. Raw image contains several unrelated foregrounds and backgrounds, and the location and size of human face are required to be specified. Namely, windows bounding multiple human faces should be detected in this step. On the other hand, the typical challenges of face detection include illumination variations, occlusions, out-of-focus blur, low image resolution, and arbitrary pose variations.

Fig. 1 Cross-Dissolving [5].

Among the state-of-the-art face detection algorithm, Liao et al. Detector [1] consisting of a hierarchy of Viola and Jones' Detector and proposed Normalized Pixel Difference (NPD) is adopted to complete the task under unconstrained situation in the proposal.

Tuning the scale of detected face region and process them to fit the face model, we try to process the images in the same scale in order to process two facial patterns properly.

Current two dimensional face morphing algorithm show the effect of merging multiple face patterns into single output, which preserve skin appearance feature. Among facial feature, skin appearance is crucial to identify human, and we assumed that the both skin and head skeleton have same importance to represent face.

We first label interest points on scaled face image by using Ramanan's Detector [10], which distribute interest point pins on whole face properly. Meshes are constructed from interest points cloud under Delaunay Triangulations connection rule, which preserves the structure of face perfectly. We put different weight on meshes creating from different face images and combine them to generate fusion mesh. Next, we warp the face by converting original mesh to fusion mesh by unitizing affine transformation. In other word, with same transformation matrix as mesh change, shape of blocks which compose single face can be converted to blocks shape which build mean face. In final step of 2D morphing, we put different weights on warped face images.

The 3D morphable model is a way to synthesis the three dimension face which is shown in Fig 3.There is a database that contains some classical face among the age, gender, height and weight of human. By using a suitable

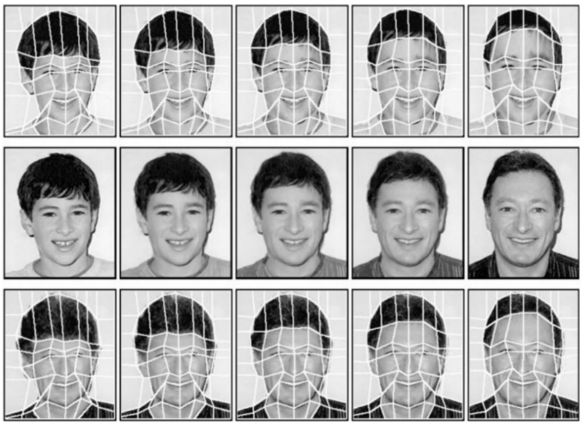


Fig. 2 Mesh Warping [5].

weighting value of the face in the database and suitable coordinate transform, the best representation of the input
2D image can be found. For our morphing of 3DMM, follow the same idea from 2D morphing. The two 3D faces are combine by choose the weighting and transform such that the result is the combination of two input image.

2. PREVIOUS AND RELATED WORK

Face morphing model is on the basis of image morphing technique, which is well developed in recent years. While image morphing techniques have been widely used in various areas, including motion morph, face recognition, animation, and movie effect re-production, this technique remains a great deal of challenges to be solved and mitigated.

First, we simply categorize previous works into 2D morph model and 3D morph model. Next, features of 2D and 3D morph models are mentioned individually, and then we compare these two different models.

Two dimensional image morphing technique was first proposed by [5]. Starting from the idea of mesh warping (Fig. 2) and cross-dissolving (Fig. 1), 2D geometric image morphing realizes the effect of combining two similar but different source images into one single result. Image warping is crucial component in [5], and some of other works apply similar skill to achieve facial beautification [6]. Although 2D image morph models have produced great success in many applications, the method still has room for improvement. For instance, global feature-based algorithm [5] in facial morphing has problem that unexpected "ghost" part of image may be generated with interpolation.

Another well-known 2D model is Active Appearance Model (AAM) [7], which contains grey-scale appearance and statistical shape model. Correlation of appearance model parameter and residual error represents relationship between training image and template model. In the testing step, residual error measured from images can predict a more accurate model parameter. As a result, a better fitting result can be generalized from AAM algorithm [7].

On the other hand, Blanz and Vetter [2] propose 3D morphable model, which is most commonly used 3D model, to automatically generate 3D face models from few photos. The idea is to produce the 3D model from an example set of 3D face models, represent both texture and shape in vector space form and control expression by forming linear combination of model. Furthermore, many state-of-the-art methods [8, 9] are inspired by 3D morphable model.

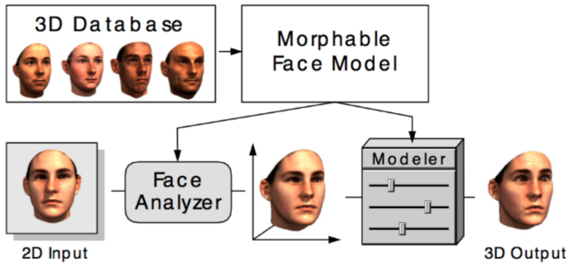


Fig. 3 3D Morphable model.

Representation powerfulness between 3D model and 2D model are compared in [3]. It is clear that 3D model has more powerful representation ability than 2D model due to the number of parameters, ability to model surface normals. Therefore, 3D morphable model is adopted in the proposal.
[Few algorithm combine 2D+3D morphable model]

3. METHODOLOGY

We argue that faces morphing should be blended both in skin appearance and shape of head skeleton, which are represented as 2D and 3D face model (Fig. 4), so that identity of blended subjects can be preserved better. In this section, face detection technique is first mentioned section 3.1. 2D blend model will be explained in section 3.2.

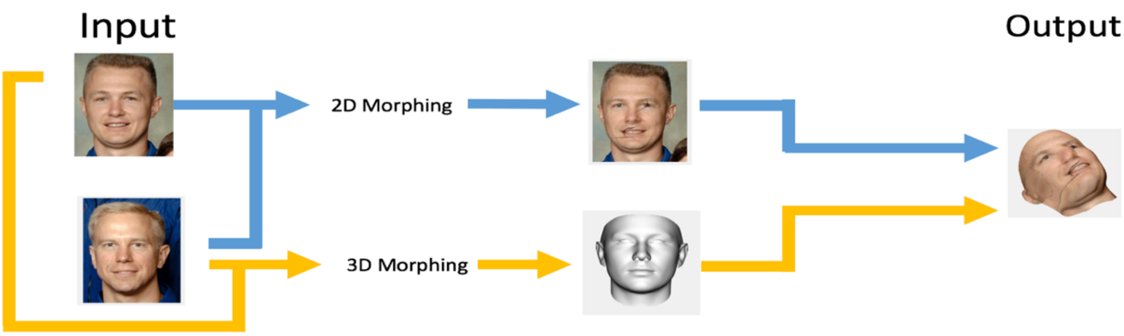


Fig. 4. Proposed 2D+3D Morphable model

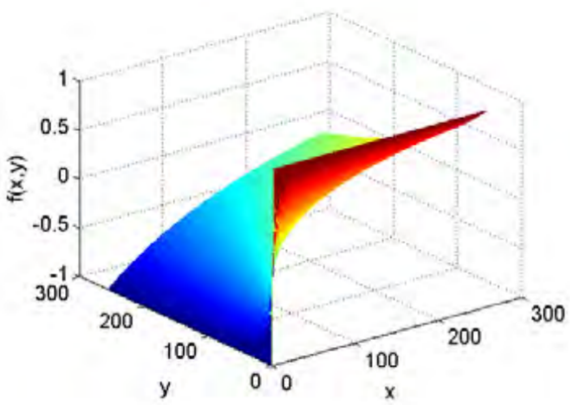


Fig. 5 NPD function *f(x,y)*.

3.1 Face Detection

Viola and Jones' detector [2] is widely used for face detection. Using Harr-like feature and cascade AdaBoost classifier, human face in single image can be detected and processed basically. However, in unconstrained scene such as arbitrary facial direction, low-resolution, and illumination changes, V. J. detector is likely to fail to work accurately.

Above problem can be solved with Normalized Pixel Difference (NPD) feature in substitution for Harr-like feature. The Normalized Pixel Difference feature between two pixels is defined as

$$f(x,y) = \begin{cases} 0 \quad , & x = y = 0 \\ \dfrac{x-y}{x+y}, & \text{otherwise} \end{cases}$$

where $x, y \geq 0$ are intensity values of the two pixels.

Invariance to scale change is one of desirable properties of NPD feature due to being expressed in form of $\Delta I / I$ (Fig. 5), which is called Weber Fraction. Furthermore, ordinal relationship between two different pixels is represented by the sign of NPD feature. In other words, the radical structure of object image can be well-extracted with NPD feature.

Liao's detector [1] applies NPD feature to realize human face detection, which can deal with unconstrained facial condition and efficiently obtain the accurate result. Therefore, Liao's detector [1] is adopted for cropping the face region from single image and creating the input for the following procedure.

3.2 2D Blend Texture

We argue that face can be represent by a few facial feature part due to human being distinguishing faces according to particular facial appearance rather than all tiny features on face. Therefore, we label face with *N* interest points by using Ramman's detector [10]. Vertices set on image *i*

$$\mathcal{V}^i = \{v_k^i\}_{k=1}^N \subset \mathbb{R}^{2\times N}$$

is generated by collecting part of facial interest points which uniformly distribute within particular region including eyes, nose, and mouth.

$$v_i^k = [x_i^k \quad y_i^k]^T$$

denotes 2D location of interest point *i* on image *k*.

As illustrated in Figure 5, our facial mesh model adopts Delaunay Triangulation which has the property, which is maximization of minimum angle among the triangles in mesh. In other word, we avoid creating "too flat" triangle in our mesh, so that face can be divided into proper size triangles. However, different interest points distribution result in different connection manner, and it cause the matching procedure to be failed. While implementing Delaunay Triangulation on every image may encounter failure for fusion, we can utilize Delaunay Triangulation on single standard image for obtaining standard connection list, which indicates the connection manner of interest points.

It comes to another problem which is different scale of image. We first convert location of vertex *j* of image *k*

$$v_j^{k'} = [x_j^{k'} \quad y_j^{k'}]^T$$

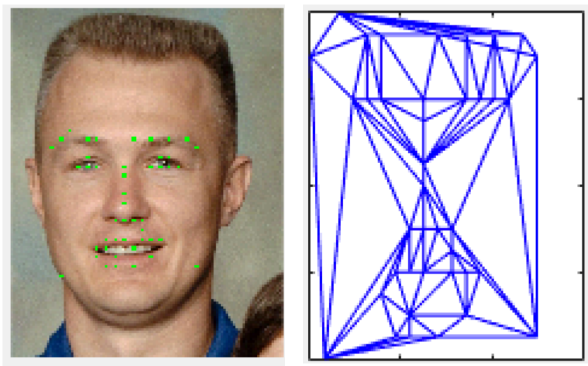
$$x_j^{k'} = \frac{x_j^k - x_{min}^k}{x_{max}^k - x_{min}^k}$$

$$y_j^{k'} = \frac{y_j^k - y_{min}^k}{y_{max}^k - y_{min}^k}$$

to resize every mesh to same scale and mesh fusion model will invariant to input image size. With same connection manner, fusion mesh can be produced by merging two or multiple mesh. We further define location of fusion point *j*

$$v_i^{fusion} = [x_i^{fusion} \quad y_i^{fusion}]^T$$

$$x_j^{fusion} = \sum_{k=1}^{n} \frac{x_j^{k'}}{n} \alpha_k$$



(a) (b)

Fig. 6 (a) Green points indicate the interest points detected by Ramman Detector (b)Facial mesh with interest points connection which follow the rule of Delaunay Triangulations

$$y_j^{fusion} = \sum_{k=1}^{n} \frac{y_j^{k'}}{n} \alpha_k$$
$$\sum_k \alpha_k = 1$$

where α_k denote weight of face *k*.

To warp the face to "mean" shape, pixels inside triangle from original image will be allocated new location. From triangle $T_m^k = \{v_{m,1}^k{}', v_{m,2}^k{}', v_{m,3}^k{}'\}$ of mesh on source image *k*, we define a matrix

$$P_m^k = \begin{bmatrix} v_{m,1}^k{}' & v_{m,2}^k{}' & v_{m,3}^k{}' \\ 1 & 1 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3}$$

which represents homogeneous location of vertices in triangle T_m^k. In the same way, from triangle $T_m^{fusion} = \{v_{m,r}^{fusion'}, v_{m,2}^{fusion'}, v_{m,3}^{fusion'}\}$ of fusion mesh, a matrix

$$Q_m^{fusion} = \begin{bmatrix} v_{m,1}^{fusion'} & v_{m,2}^{fusion'} & v_{m,3}^{fusion'} \\ 1 & 1 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3}$$

is defined as homogeneous location of vertices in triangle T_m^{fusion}.

Next, for triangle m warping procedure, we compute transformation matrix

$$H_m = Q_m^{fusion} P_m^{k^{-1}}$$

which maps pixels inside T_m^k to ones inside from T_m^{fusion}. In order to warp single face, every triangle of it will change to new shape after computation of own transformation matrix. Furthermore, inverse warping, which means searching pixel value in original image to fill in destination image, are adopted to avoid missing pixel value caused by using forward warping. Besides, the computation complexity of warping single face is O(N).

Finally, we blend multiple face using different weight on different source image and then produce blended image

$$I^{fusion} = \sum_{k=1}^{n} \frac{I^k}{n} \alpha_k$$

where $\sum_k \alpha_k = 1$. We set the weights on wrapped images as same as the weights on meshes to merge multiple face better.

3.3 3D Morphable Model

Our method for finding 3D face model is based on the algorithm in [11], which can be divided into two steps. First, find the pose of the face. Second, find the shape of the face.

In order to find the pose of the face, we begin by describing the scaled orthographic projection (SOP), which is a projection from 3D point $v = [v_x, v_y, v_z]^T$ to the 2D point $X = [x, y]^T$. Assuming the 3D point set have similar distance to the 2D plane, we can represent the SOP by,

$$X = SOP(v, s, R, t) = s\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} Rv + st.$$

where s is the positive scaling constant, R is the 3D rotational matrix, and t is a 2D translation vector.

3D morphable model (3DMM) is the combination of the principal model and the mean model. We define a matrix

$$M = [v_{x1}, v_{y1}, v_{z1}, v_{x2}, v_{y2}, v_{z2} \dots, v_{xN}, v_{yN}, v_{zN}]^T,$$

which represents a sequence of N vertices.
Therefore, the 3DMM can be represented as

$$s(\alpha) = P\alpha + \bar{s}$$

where $P = [M_1, M_2, M_3 \dots, M_S]$ denote the concatenate of S principal models, $\alpha = [\alpha_1, .. \alpha_S]^T$ is the weighting value of each principal model and $\bar{s}$ is the mean model.
Fitting face model is then recognized to find a tuple

$$(\alpha, s, R, t) = \operatorname{argmin} \Sigma_{i=1}^{L} |X - SOP(s(\alpha), s, R, t)|^2,$$

which minimize the 2-norm error between the face after scaled orthographic projection and the L pixels, which are the landmarks from Ramman detector [10].

In order to solve the multivariable optimization problem, we first optimize the SOP parameters (s, R, t) by assuming that α equals zero. In other words, we find the optimized affine transform for the mean face. The procedure is similar to POS algorithm that can be found in [12].

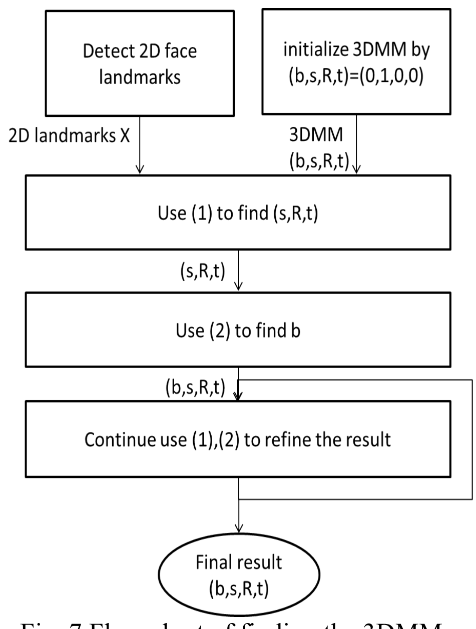
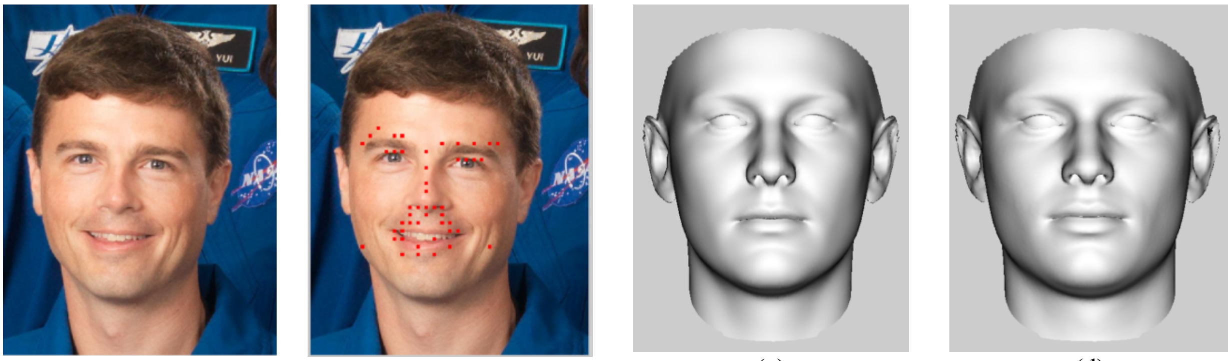


Fig. 7 Flow chart of finding the 3DMM.



(a) (b) (c) (d)

Fig. 8 Finding the 3DMM. Left-most (a) is input image, and interest points are produced with Ramman Detector [10] shown in (b). (c) is initial 3DMM, and then optimize 3DMM to generate (d).

First, we solve the 2D correspondence of the equation

$$\begin{pmatrix} v_1 & 0 & & v_L & 0 \\ 1 & 0 & & 1 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & & 0 & 0 \\ 0 & v_1 & & 0 & v_L \\ 0 & 1 & & 0 & 1 \end{pmatrix} k = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_L \end{pmatrix},$$

where k=$(k_1,..k_8)^T$ is a $\mathbb{R}^8$ vector that can be solved by using linear least square. Next, the SOP parameters s and t can be computed by

$$\mathrm{s} = \frac{||\mathrm{r}_1|| + ||\mathrm{r}_2||}{2}, \qquad \mathrm{t} = \left[\frac{\mathrm{k}_4}{\mathrm{s}}, \frac{\mathrm{k}_8}{\mathrm{s}}\right]^{\mathrm{T}}$$

where $\mathrm{r}_1 = [k_1, k_2, k_3]$ and $\mathrm{r}_2 = [k_5, k_6, k_7]$.

For rotation matrix R, we perform the singular value decomposition on the 3 by 3 matrix

$$\begin{pmatrix} r_1 \\ r_2 \\ r_1 \times r_2 \end{pmatrix} = USV^T.$$

The rotation matrix R will be UV^{T}. While the parameters (s, R, t) are not the optimal solution to fit a given 2D face, it is very close to the optimal one and we can utilize these values as an initial condition to do another optimization procedure with a good initial condition of the weighting α.

The next step is to find the weighting α. We solve the least square problem

$$\mathrm{X}' = \mathrm{sR}(\mathrm{P}\alpha + \bar{\mathrm{s}}) + st$$

by applying the SOP parameters which have been computed into the SOP and 3DMM.

After rearranging the equation, we get the linear equations

$$\mathbf{Cb} = \mathbf{h}$$

where $\mathrm{C} = [\mathrm{C}_1, \mathrm{C}_2, \dots \mathrm{C}_\mathrm{S}] \in \mathbb{R}^{2\mathrm{L}\times\mathrm{S}}$, b$\in \mathbb{R}^{\mathrm{S}}$ is the optimal weighting vector, and $\mathrm{h} \in \mathbb{R}^{2\mathrm{L}}$.

$$\mathrm{C}_{2\mathrm{i}-1} = s(R_{11}P_{i1}^T + R_{12}P_{i2}^T + R_{13}P_{i3}^T)$$

$$\mathrm{C}_{2\mathrm{i}} = s(R_{21}P_{i1}^T + R_{22}P_{i2}^T + R_{23}P_{i3}^T)$$

$$\mathrm{h}_{2\mathrm{i}-1} = \mathrm{x}_\mathrm{i} - s(R_1\bar{s} + t_1)$$

$$\mathrm{h}_{2\mathrm{i}} = \mathrm{y}_\mathrm{i} - s(R_2\bar{s} + t_2)$$

where R_{ij}, $\mathrm{R}_\mathrm{i}$denote the entry of the i^{th} column and j^{th} row of the rotation matrix and the i^{th} column of the rotation matrix. $(\mathrm{x}_\mathrm{i}, y_i)$ is the i^{th} landmark in the 2D face, and $\mathrm{t} = (\mathrm{t}_1, \mathrm{t}_2)^{\mathrm{T}}$ is the translation vector.

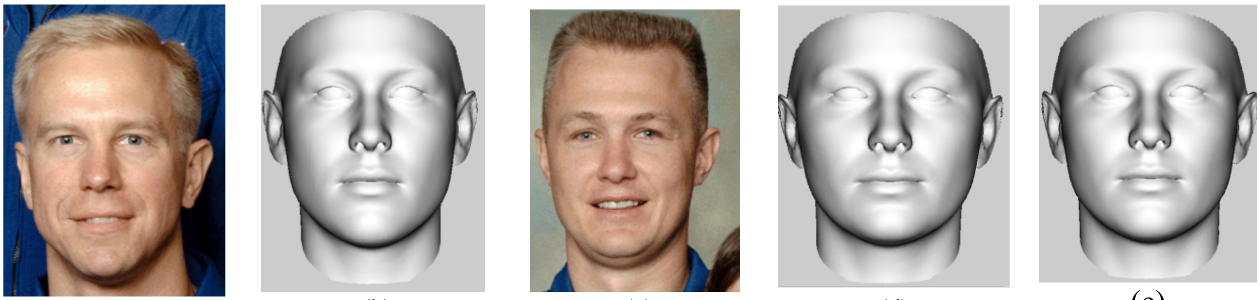
After computing the first trial of the optimal (b, s, R, t), we can further optimize them by using the 3DMM with weighting b to generate L new vertex. We follow the same procedure described above to find the new (s, R, t) sequence and use them to find the more optimal weighting b.

In order to fit the model to the 2D face, we do not only consider to match the landmark, but also match the edge on the 2D face using similar idea as iterated closest point (ICP).

As illustrated in Figure 7, we summarize the procedure of finding the optimal 3D face. Figure 8 shows that case of fitting the 3DMM. Note that the face in is boarder and more similar to the man in Fig. 8(a) than the mean face in Fig. 8(c).

3.4 3D Morphable Model Rendering

The rendering of the 3DMM is simple. First of all, we perform the SOP on the model by the optimized (s, R, t). Next, each vertex will map to the 2D plane. If the position of the projected vertex overlap with the pixels in the 2D face image, we set the color of the vertex to the color of the overlapped pixel. After the initialization, we use the z-buffer to plot the result.



(a) (b) (c) (d) (e)

Fig. 11 A instance of Morphing two 3DMM. (a) (c) is two source images. Corresponding 3DMM is shown on (b)(d). The last image is the morphing result.

3.5 3D Morphable Model Merging

What we then concern about is how to find the 3DMM for the 2D morphing result. Because we do not want to lose the skeleton information on the 3DMM of each person, the result generated from above finding 3DMM algorithm using 2D morphing image as input is undesirable. In the case, we interpolate the parameters $(\mathrm{b}, \mathrm{s}, \mathrm{R}, \mathrm{t})$ to preserve the skeleton information. Intuitively, we expect the processed face is always in front angle whether the subject looks rightward or leftward. We preserve the properties of each parameter the result parameter $(\hat{\mathrm{b}}, \hat{s}, \hat{R}, \hat{t})$

$$\hat{\mathrm{b}} = \frac{b_1 + b_2}{2}$$

$$\hat{\mathrm{s}} = \sqrt{s_1 \cdot s_2}$$

$$\hat{\mathrm{R}}^2 = \mathrm{R}_1 R_2$$

$$\hat{\mathrm{t}} = \frac{t_1 + t_2}{2}$$

where $(\mathrm{b}_1, s_1, R_1, t_1)$ and $(\mathrm{b}_2, s_2, R_2, t_2)$ are the parameters generated from the procedure in 3.3 with two 2D face f_1 and f_2 respectively. We show a case of morphing two given 3DMM in Fig. 9. Note that the width of the morphing 3DMM result is between the thinner and fatter mans of the input images.

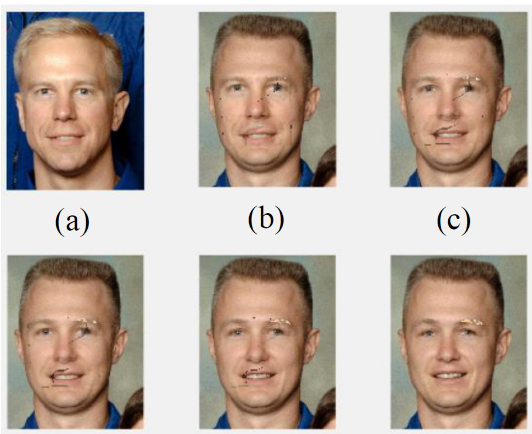
4. EXPERIMENT

We expect our morphing model is one-click program, which means our input is single photo and output is blended face 2D and 3D model merging two or multiple persons. Three different group photos from NASA astronaut group is used, and we will build the 2D+3D morphing model for preserving and combining feature of multiple persons in the experiment

4.1 2D Face Morphing

For the purpose of merging two different source images, we build fusion mesh by source meshes which is consist of interest points detected by Ramman Detector [10]. More detail experiment result and concept of Ramman Detector are described in [10].

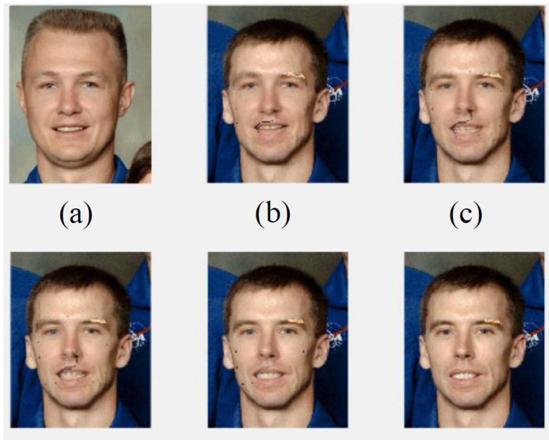
To clearly demonstrate the effect of 2D Face Morphing, only two of persons in group photo will be



(a) (b) (c)

(d) (e) (f)

Fig. 9 Blend images with different weights. (a)(f) are source images selected from group photo (b) α_2= 0.2 (c) α_2= 0.4 (d) α_2= 0.6 (e) α_2= 0.8



(a) (b) (c)

(d) (e) (f)

Fig. 10 Blend images with different weights. (a)(f) are source images selected from group photo (b) α_2= 0.2 (c) α_2= 0.4 (d) α_2= 0.6 (e) α_2= 0.8

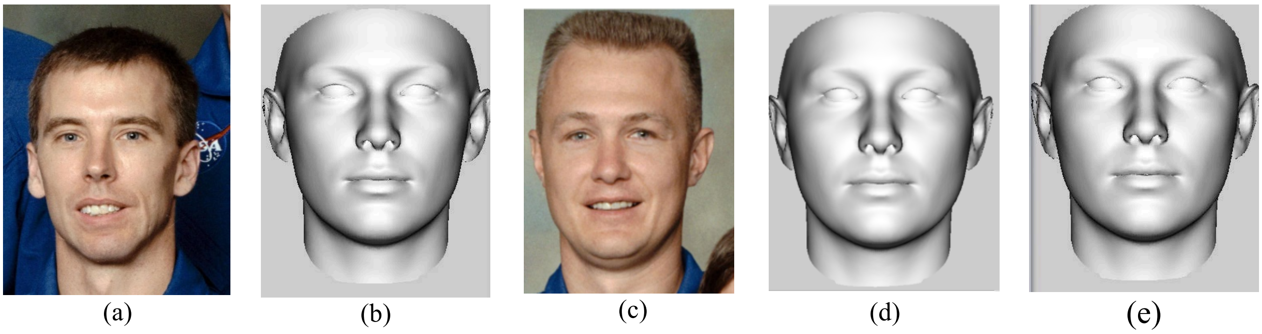


Fig. 12 The other instance of Morphing two 3DMM. (a) (c) is two source images. Corresponding 3DMM is shown on (b)(d). The last image is the morphing result.

used as input images. As illustrated in Figure.7,8 , we first set α_k = { 0.2, 0.4, 0.6, 0.8 }, which is weight of images k.

Notice that we focus on blending center of face and preserving other parts instead of merging whole face and hairstyle. In the 4.3 combination section, $\alpha_k = 0.5$ which put same importance on both source images will be used.

4.2 3D Face Morphing

As illustrated in Fig. 11 and Fig. 12, we show 3D Face morphing algorithm. Notice that while it show similar in 3D morphable model, the width of the morphing 3DMM result in Fig. 11 is between the thinner and fatter mans of the input images.

4.3 2D+3D Morphable model

As illustrated as Figure 4, the proposed algorithm combines both 2D and 3D model in final step so as to completely generate morphing face of multiple sources image, where morphing skin appearance and head skeleton information are generated from 2D and 3D model. We show morphing skin, mixed head skeleton, and our result in Fig. 13. We can observe that 2D, 3D, and 2D+3D morphing results are different from each other in spite of one of source images are same.

CONCLUSION

In this paper, we merge multiple human face by way of mixing 2D and 3D morphing algorithm. Features implied identity including skin appearance and head skeleton are well preserved and blend under proposed framework. While this still encounter some challenges (e.g. missing pixel value in 2D morphing and more suitable 3D model), proposed algorithm can be applied on animation, gaming and other applications.

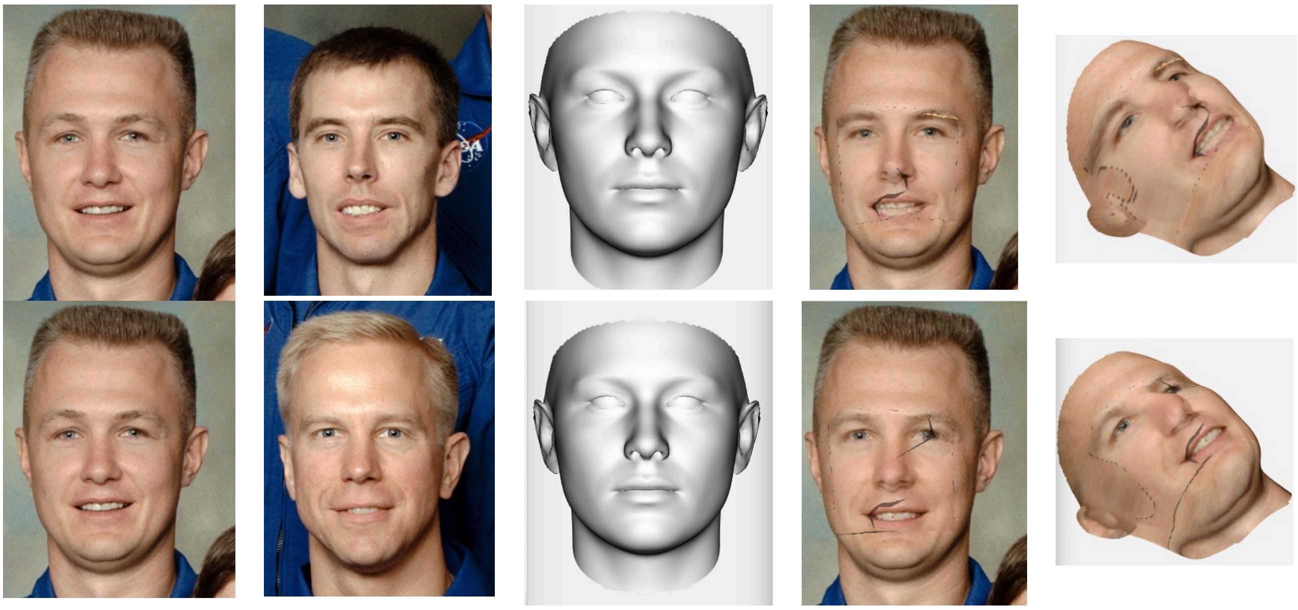


Fig. 13 First and second row indicate different instance. First two columns are sources images. Third column is 3D morphing results. Fourth column is 2D morphing results. Last column are proposal algorithm result.

REFERENCES

[1] S. Liao, A.K. Jain, S.Z. Li, “A Fast and Accurate Unconstrained Face Detector”, PAMI 2016

[2] V. Blanz, T. Vetter ,“A Morphable Model For The Synthesis Of 3D Faces”, SIGGRAPH 1999,

[3] I. Matthews, J. Xiao, S. Baker, “2D Deformable Model v.s. 3D Deformable Model”, IJCV 2007

[4] J. Xiao, S. Baker, I. Matthews, T. Kanade ,“Real-time combined 2D+ 3D active appearance models”, CVPR 2004

[5] T. Beier, S. Neely, “Feature-Based Image Metamorphosis”, SIGGRAPH 1992

[6] T. Leyvand, D. Cohen-Or, G. Dror, D. Lischinski. “Data-Driven Enhancement of Facial Attractiveness”, SIGGRAPH 2008,

[7] T. Cootes, G. Edwards, and C. Taylor, “Active appearance models”, PAMI 2001,

[8] J. Thies, M. Zollhöfer et al., “Face2Face: Real-time Face Capture and Reenactment of RGB Videos”, CVPR 2016,

[9] A. Bas, W. A. P. Smith, T. Bolkart, S. Wuhrer, “Fitting A 3D Morphable Model to Edges: A Comparison between Hard and Soft Correspondences”, arXiv:1602.01125 [cs.CV]

[10] X. Zhu, D. Ramanan, "Face detection, pose estimation and landmark localization in the wild", CVPR 2012.

[11] W. A. P. S. Anil Bas, Timo Bolkart, Stefanie Wuhrer, "Fitting a 3D Morphable Model to Edges: A Comparison Between Hard and Soft Correspondences," *IEEE International Conference on Image Processing,* 2016.

[12] D. F. Dementhon and L. S. Davis, “Model-based object pose in 25 lines of code,” Int. J. Comput. Vis., vol. 15, no. 1-2, pp. 123–141, 1995.

[13] http://www.spacefacts.de/groups/large/english/nasa-18.htm , NASA astronaut group photo

[14] https://github.com/waps101/3DMM_edges, 3DMM_edges