Surface Detail Capturing for Realistic Facial Animation

Pei-Hsuan Tu, I-Chen Lin, Jeng-Sheng Yeh, Rung-Huei Liang, and Ming Ouhyoung

Communication and Multimedia Laboratory, Department of Computer Science and Information Engineering "National Taiwan University"

E-mail: {keico, ichen, jsyeh, liang}@cmlab.csie.ntu.edu.tw; ming@csie.ntu.edu.tw

Revised June 29, 2004.

Abstract In this paper, a facial animation system is proposed for capturing both geometrical information and illumination changes of surface details, called expression details, from video clips simultaneously, and the captured data can be widely applied to different 2D face images and 3D face models. While tracking the geometric data, we record the expression details by ratio images. For 2D facial animation synthesis, these ratio images are used to generate dynamic textures. Because a ratio image is obtained via dividing colors of an expressive face by those of a neutral face, pixels with ratio value smaller than one are where a wrinkle or crease appears. Therefore, the gradients of the ratio value at each pixel in ratio images are regarded as changes of a face surface, and original normals on the surface can be adjusted according to these gradients. Based on this idea, we can convert the ratio images into a sequence of normal maps and then apply them to animated 3D model rendering. With the expression detail mapping, the resulted facial animations are more life-like and more expressive.

Keywords facial animation, facial expression, deformations, morphing, bump mapping

1 Introduction

With the accumulation of pioneering work in face synthesis for two decades, the research interests of synthetic face increase rapidly. Nowadays, synthetic faces are extensively used in various kinds of media. However, it is still a labor-intensive work for animators to generate realistic facial animation. That is because a face is the most expressive and variable part in one's appearance, and each detail of a face may have revealing meanings for us.

Recently, the motion capture technique is widely used in speeding up the process of facial animation production. Dozens of feature markers are placed on a subject's face and the motion capture technique is used to track 2D or 3D motion trajectories of markers. These trajectories provide a paradigm to adjust face models. In our previous work[1], we developed an inexpensive and accurate 3D motion tracking method that can track fifty-five markers and now it is further extended to three hundred markers. With such a large quantity of motion data, our synthetic face can mimic large portions of external shape variations of a real face. However, subtle portions, such as wrinkles or creases, whose variations are smaller than a marker's diameter, are difficult to be acquired by motion capture techniques. In the cutting-edge movie industry, specific 3D laser scanners with very high resolutions are employed to capture subtle wrinkles and creases on a face. Such an approach provides convincing results but it is too expensive and infeasible for real-time or near real-time facial animation.

The goal of the proposed work is to efficiently capture expression details for real-time animation without heavy user intervention. Figs.1 and 2 show the processes of our system. In the data capturing stage, we are inspired by Liu *et al.*'s work[2] and capture expression details from intensity ratios. To reconstruct approximate face shapes, we utilize the captured markers' motion trajectories based on our previous tracking technique. The subtle wrinkles and creases uncovered by markers' motion are tackled by animated texture sequences. For 3D facial animation, it is insufficient to animate expression details by textures only since the intensities of expression details can change dramatically according to different incident and reflective angles. To handle this, in the proposed work, we estimate the detailed normal variations on face surfaces from the gradients of intensity ratios caused by facial expression, and a hardware-supported bump mapping can be applied to render such detailed normal variations. With this framework, we can display delicate 3D facial expressions without additional complexities. Therefore, the proposed method can efficiently animate synthetic faces with subtle expressions in both 2D and 3D situations.

The paper is organized as follows. Some state-of-the-art researches are surveyed for comparison in Section 2. In Section 3, we will first explain the

[*] Correspondence

facial motion capture stage of the whole process. Section 4 describes how the expression detail is captured and how it is converted into normal maps for the bump mapping. The face synthesis stage for our system is described in Section 5. Some results are shown in Section 6. Finally, we conclude this paper with a discussion of the proposed system and future research directions in Section 7.

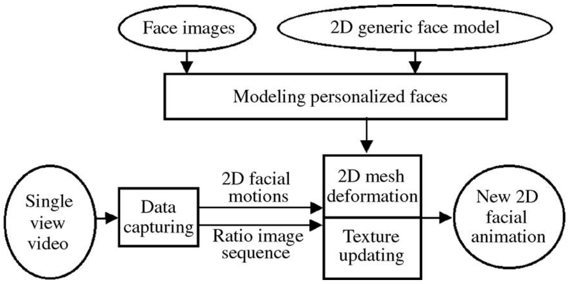

Fig.1. Block diagram of the proposed 2D Facial Animation System.

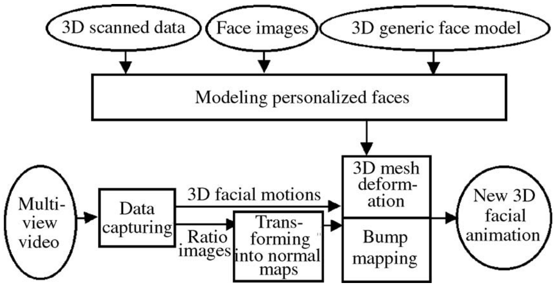

Fig.2. Block diagram of the proposed 3D Facial Animation System.

2 Related Work

For facial animation, most of researches focused on facial shape estimation. The muscle-based approach is derived from the concept of anthropotomy, where external facial shapes are varied due to the interaction of internal muscles and skin tissues. Platt *et al.*[3] and Waters[4] proposed pioneering face models with physical muscles. Even though muscles and skin tissues are simulated in muscle-based facial animation, these models are still having troubles to deal with subtle wrinkles and creases. The performance-driven facial animation method proposed by Williams[5] provide a direct and effective method to make facial animation from captured data of performers. In the research of Guenter *et al.*[6], they generated video-realistic 3D facial animations by capturing both geometric shape variations and facial textures simultaneously from multiple-view videos. Their framework is similar to that of our proposed work. Captured texture sequences in Guenter *et al.*'s work can only be applied to the same subject. By contrast, we extract expression details from video clips and therefore, our captured expressions can be applied to others' faces conveniently.

Image-sample based methods interpolate image segments to re-synthesize facial animation. The "video rewrite" proposed by Bregler *et al.*[7] generates facial animation by combining facial image samples according to input phonemes. Cosatto *et al.*[8] further decomposed the samples into smaller parts and constructed a sample space for efficient synthesis. Ezzat *et al.*[9] proposed a training model to synthesize novel facial motion sequences from basis facial images. Since the image-sample-based methods directly integrate with facial images where subtle wrinkles are included, this kind of method could be the most realistic approach for facial animation, but viewing directions are limited and the sample data is difficult to be applied to other faces.

For subtle expressions such as wrinkles or creases, Wu *et al.*[10] synthesize facial wrinkles with a bio-mechanical skin model. This approach can perform efficiently in real-time but it is not intuitive to control the appearance of wrinkles by physical parameters. Tiddeman *et al.*[11] proposed a wavelet-based method to synthesize face aging by prototype images. Liu *et al.*[2] proposed the expression ratio image that clones subtle expressions from a single facial image. In Bando *et al.*'s work[12], they presented a simple method to control synthetic wrinkles.

3 Motion Capture

Before extracting subtle expression details, the approximate face shape or outlines have to be recovered. Colorful markers are placed to features of a human face. It is much easier and more accurate to track movement of feature points with these markers. We use a digital video camera to capture a subject's expressions. For 2D facial motion estimation, a single view image is sufficient. But for 3D position reconstruction, multiple view images are required. Instead of using three cameras, we place two mirrors next to the subject's face and use only one camera to capture the multiple view images. The two mirrored images can be regarded as flipped images taken by two virtual cameras (see Fig.3).

In our system, a semi-automatic approach is adopted to track the markers on the face. For 2D facial motion estimation, after locations of markers in each frame are determined, we obtain the difference vectors between the corresponding markers of the first frame and the remainder frames. For 3D facial motion estimation, we adopt our previous method[1]. After determining the 2D projected positions of each marker and the correspondence in front and mirrored views, the 3D positions of markers in each frame can be reconstructed.

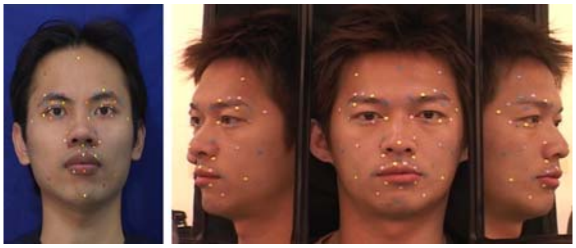

Fig.3. Image data captured by a digital video camera. Left: Single-view image (480 × 720 pixels) and 61 markers are placed. Right: Multi-view image (720 × 480 pixels) and 49 markers are placed.

4 Expression Detail Capture

4.1 Ratio Image Sequence Extraction

In addition to geometric data of facial expressions, we refer to Liu *et al.*'s method[2] to capture subtle illumination changes of expressive faces by ratio images. Instead of a static image, ratio images are continuously evaluated from video clips in our system. To extend the expression ratio images for animation, the above-mentioned semi-automatic marker tracking is employed to replace the manual feature point marking in Liu *et al.*'s[2] work.

Since the first expression in a video clip is required to be neutral, the first frame is used as a neutral face image and is denoted by F_0. Remainder frames are used as expressive face images and are denoted by F_i, $1 \leqslant i < n$, where n is the total number of frames. In Liu *et al.*'s approach, a neutral face is warped to align with an expressive face, and then a ratio image is computed pixel by pixel. However in our situation, the ratio images extracted from video clips will be used for dynamic textures (in 2D facial animation) and for the dynamic bump mapping (in 3D facial animation). According to the convention of animation, the texture coordinates are fixed in the initial stage, which is the neutral face in our case. To simplify the work of applying ratio images described in the next section, we warp all expressive faces to align with the neutral face before computing intensity ratios.

Locations of markers tracked in each frame are where the facial features are. We compute the difference vectors between the feature positions of F_0 and F_i and then move the features of F_i along the difference vectors. For other pixels that are not feature points in the image F_i, an RBF (Radial Basis Functions) data scattering method is utilized to conjecture their displacement.

The RBF is well known for its interpolation capability whether in 2D or 3D, so it is repeatedly used in various steps of our system, such as image warping, 2D and 3D face modeling, texture mapping and bump mapping. A method based on the RBF is adopted to represent the influence of constrained points. We chose the radial basis function as $\phi(r) = e^{\frac{-r}{k}}$, where k is a user-defined constant. The data scattering function is of the form:

$$f(p) = \sum_i c_i \phi(\|p - p_i\|) + Mp + t$$

where p_i is the constrained feature vertex i, low-order polynomial terms $M^t = [a, b]$ and t are added as an affine basis. To determine the unknown coefficients c_i and the affine components M and t, we must solve a set of linear equations that includes $u_i = f(p_i)$, the constraints $\sum c_i = 0$ and $\sum c_i p_i^t = 0$. In general, if there are n feature point correspondences, we will have $n + 3$ unknowns and $n + 3$ equations with the following form (in the 2D case):

$$\begin{bmatrix} \cdots & \cdots & \cdots & \cdots & p_{1x} & p_{1y} & 1 \\ \vdots & e^{-\|p_j - p_k\|/64} & \cdots & \cdots & p_{2x} & p_{2y} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \cdots & \vdots & p_{nx} & p_{ny} & 1 \\ p_{1x} & p_{2x} & \cdots & p_{nx} & 0 & 0 & 0 \\ p_{1y} & p_{2y} & \cdots & p_{ny} & 0 & 0 & 0 \\ 1 & 1 & \cdots & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \\ a \\ b \\ t \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where $1 \leqslant j,\ k \leqslant n,\ P_i = (p_{ix}, p_{iy})$.

4.2 Conversion from Ratio Images to Normal Maps

For 3D facial animation, animating expression details by dynamic textures is insufficient since the intensities of expression details can change according to different incident and reflective angles. In our system, we adopt the bump mapping to present subtle expressions instead, because rendering 3D models with the bump mapping can make the animated faces reveal subtle surface details without additional geometric complexities. How to generate normal maps from ratio images is described in the following.

A ratio image is defined by dividing colors of an expressive face by those of a neutral face. In our observation, pixels with ratio value smaller than 1.0 are where wrinkles, creases or indentations of the expressive face appear (see Fig.4).

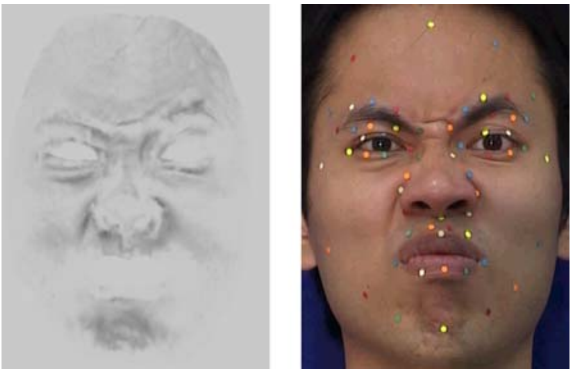

Fig.4. Left: Ratio image after converting ratios to intensity for display purpose. Right: Facial creases with lower intensity are usually wrinkles, creases or indentations on the surface.

The smaller the ratio is, the deeper the point on the surface is. Therefore, variations of ratio values would be regarded as subtle changes on a face and we can construct a surface $Sr(u,v)$ according to the ratios to simulate undulations on the expressive face surface (as shown in Fig.5).

For $S_r(u,v)$, the gradient of S_r at coordinates (u,v) is defined as the vector[13]:

$$\nabla \boldsymbol{S}_r(u,v) = \begin{bmatrix} \dfrac{\partial S_r}{\partial u} \\ \dfrac{\partial S_r}{\partial v} \end{bmatrix}$$

The adjusted normal vectors will be:

$$\boldsymbol{N}_{new} = Normalize\left(\boldsymbol{N} + \left(\boldsymbol{U} \times \frac{\partial S_r}{\partial u}\right) + \left(\boldsymbol{V} \times \frac{\partial S_r}{\partial v}\right)\right)$$

$\boldsymbol{N}$ is the original vertex normal, $\boldsymbol{U}$ is the unit vector in the direction of increase of the u texture, and $\boldsymbol{V}$ is the cross product of $\boldsymbol{N}$ and $\boldsymbol{U}$. The local coordinate defined by them is the tangent space of each point on the surface (see Fig.6).

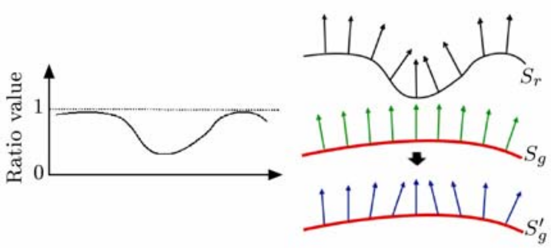

Fig.5. Left: Ratio image can form a pseudo surface. Right: S_r is constructed according to ratios in the ratio image and the original normals on the face surface. (S_g, reconstructed by retargeting of facial motion trajectories, can be adjusted.)

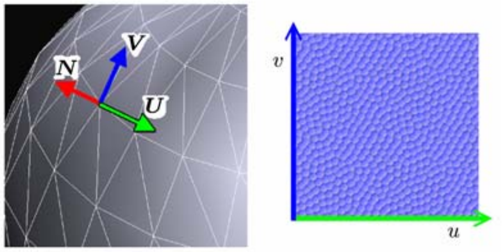

Fig.6. Left: Tangent space with vertex normal $\boldsymbol{N}$ (red), $\boldsymbol{U}$ (green) and $\boldsymbol{V}$ (blue) axes. Right: Texture coordinate of a normal map.

In other words, we construct a pseudo surface and conjecture the orientation of the normal vector adjustment according to the gradients of intensity ratios. For every pixel in a ratio image, we estimate new normal vectors in the tangent space and convert the X, Y, Z components into R, G, B color values; hence a normal map is generated. After repeating the previous process frame by frame, we can convert all ratio images extracted from video clips into normal maps.

5 Synthetic Face

5.1 Face Modeling

All personalized face models are deformed from the generic face models. In the 2D case, a generic face model is generated by applying Delauney triangulation to a set of points. There are two major stages in our 3D face modeling process. First, users manually specify the corresponding features in the scanned face versus the generic face. After computing the difference vectors between the features of the two faces, we adopt a 3D RBF data scattering method to find out the displacements of other vertices in the generic model. After the RBF deformation, a preliminary modeling is complete (Fig.7(b)). Because the number of corresponding features specified by users is limited, the deformed model cannot completely fit the surface of the scanned face. For ensuring a more precise surface match, cylindrical projection[14] is used to project each vertex in the deformed face model onto the scanned face. A vertical line through the centroid of the model is taken as a cylindrical projection centerline. A ray perpendicular to the projection centerline passes through each vertex in the deformed model and intersects with scanned data. Then we move each vertex to the intersection of a centerline and the surface of scanned data. However, since the scanned data may be registered from scan results of previous passes, a ray can intersect with more than one surface of scanned data. Hence we average these intersections, and the vertex passing through the ray is moved to the averaged 3D position. After projecting each vertex, the model matches the scanned data completely and the second stage of face modeling is done (see Fig.7(c)).

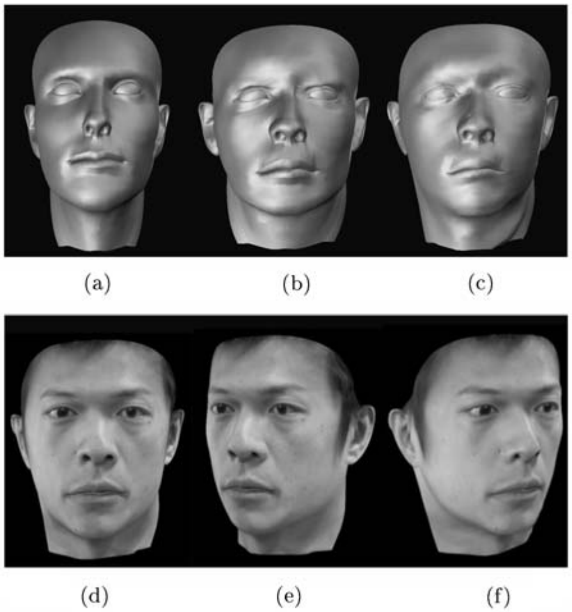

Fig.7. Reconstructed 3D face model and texture mapping. (a) Generic model. (b) RBF-deformed model. (c) Deformed model after the cylindrical projection. (d)–(f) are synthetic faces from different views.

5.2 Facial Animation

5.2.1 *Facial Motion Retargeting*

After the personalized face models are constructed, facial motions captured from video clips are applied to control points on the face model. When the displacements of all control points are determined, the face models can also be deformed by the RBF data scattering method. Once we repeat the previous process frame by frame, we can generate facial animations according to the estimated facial motion data. Up to now, the resulting animations are generated only by the geometric-warping-based approach. Although expressions are generated, they are still somewhat untruthful. The major reason is that the animations do not fully represent the expression details such as wrinkles and facial creases. In the following, we introduce how to map the captured expression details into 2D and 3D animated faces.

5.2.2 *Expression Detail Mapping*

Owing to the hardware texture mapping capability in most graphics cards today, we use a mesh-based approach for geometric warping and texture animation for expression detail mapping. The ratio image sequence extracted in Section 3 is used to update illumination changes of the subject's expression. Given a face image B and a ratio image sequence R_i, $0 \leqslant i < n$, where n is the number of ratio images, there are three steps used to explain how the texture animation process works in 2D.

Step 1. Compute the difference vector between the corresponding feature positions of B and R_0. Align R_0 with B through RBF warping and there will be a mapping between B and R_0. Create a table T the size of image B and record correspondences between B and R_0 in T. For each pixel of B, we can find a corresponding pixel in R_0 by consulting T.

Step 2. Get a result image $B'(u, v) = R_0(T(u, v)) \times B(u, v)$.

Step 3. Let B' be a new texture image.

Because all ratio images are using the same expression, the correspondence between B and all ratio images must be the same. The table T is just constructed once and can be used recurrently. By repeating Step 2 and Step 3, ratio images are calculated one by one and new texture images are thus produced accordingly. Therefore, texture animation is generated and the expression details extracted from video reappears in the face image B with texture animation.

Expression detail mapping in 3D is similar to that in 2D. We adopt the bump mapping technique to apply normal maps to 3D face model rendering. Besides, a ratio image is captured every 5 frames of the video. Therefore, an interpolation of cosine function is chosen to generate in-between ratio images or normal maps in key-framing animation[27].

6 Results [①]

In a 2D case, sixty-one colorful markers are placed to a performer's face and we capture his expressions by a digital video camera. The results in 2D are shown in Figs.8 and 9.

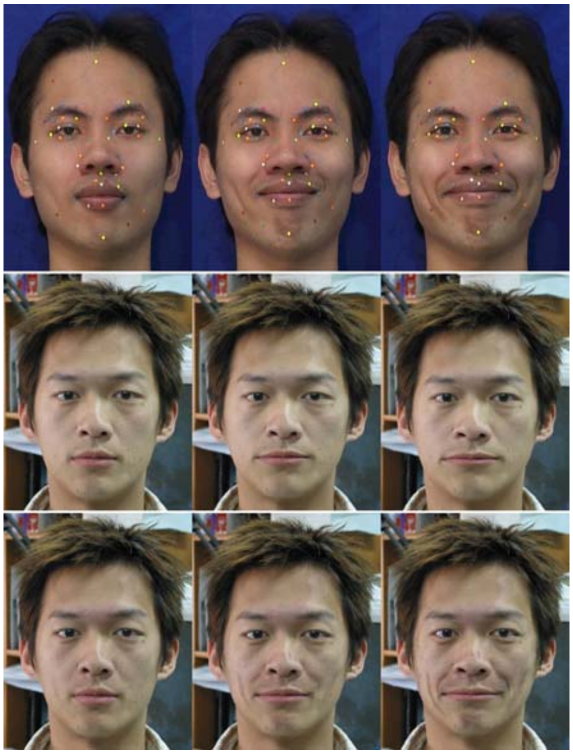

Fig.8. Images of "smile". The first row: Captured frames in the video. The second row: Synthesized results without the expression detail mapping. The third row: Corresponding synthesized results with the expression detail mapping.

In a 3D case, forty-nine colorful markers are placed to a performer's face. We place two mirrors next to the subject's face and use one digital video camera to capture his expressions by capturing multiple-view images which is an approach used by Lin *et al.*[1,15]. The results in 3D are shown in Figs.10 and 11.

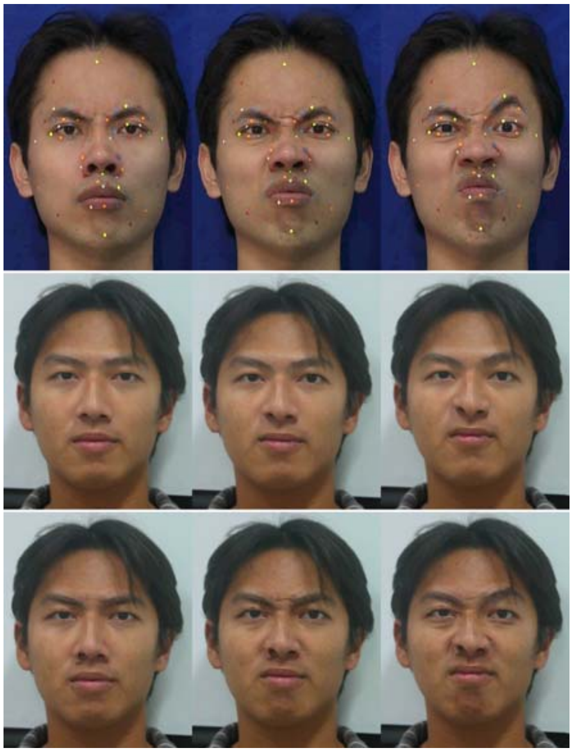

Fig.9. Images of "anger". The first row: Captured frames in the video. The second row: Synthesized results without the expression detail mapping. The third row: Corresponding synthesized results with the expression detail mapping.

7 Conclusion and Future Work

In this paper, a complete system for both 2D and 3D realistic facial animation is proposed. Besides geometric changes of a face, the detailed illumination changes of human's expression are also modeled. Instead of calculating detailed geometric variations, which is infeasible for real-time animation, we represent subtle geometric changes of expressive face surface by normal maps. The gradients of intensity ratios are used to evaluate the variations of the normal vectors on the surface of an input 3D face model. By animating a face with these visually important details of facial expressions, the resulted facial animations are much more realistic, life-like and expressive. Moreover, the data captured from video clips can be applied to different face models.

[①] The demo video is located in our web page: http://www.cmlab.csie.ntu.edu.tw/~keico/jcst/edm.html

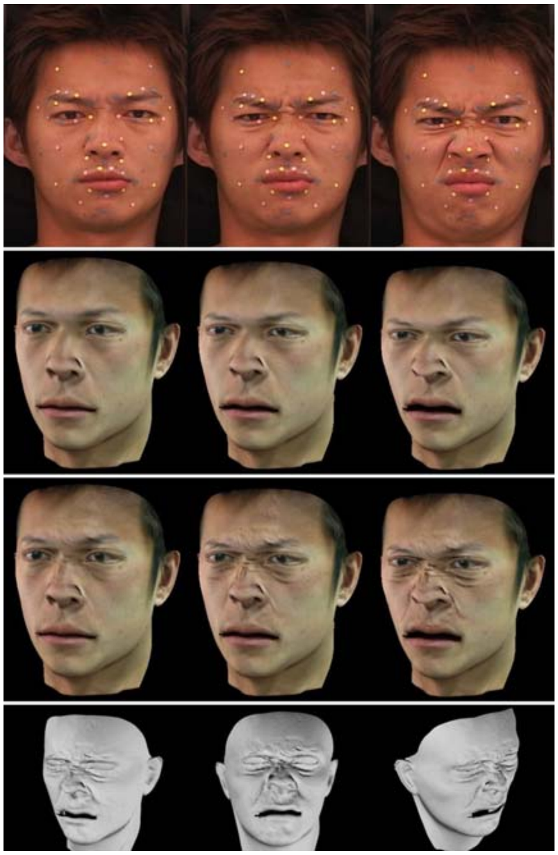

Fig.10. Images of "anger". The first row: Captured frames in the video. The second row: Synthesized results without the expression detail mapping. The third row: Synthesized results with the expression detail mapping. The fourth row: Close-up shots of the synthesized faces without texture from different views.

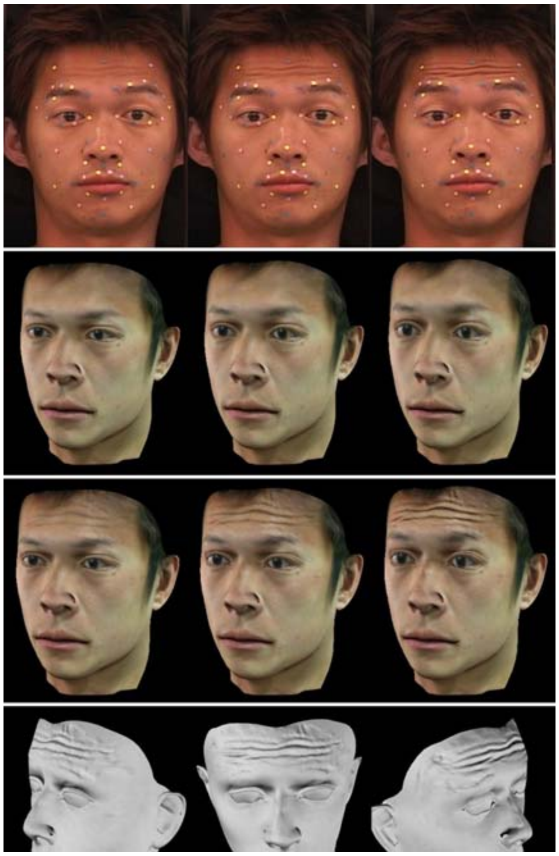

Fig.11. Images of "raising-eyebrow". The first row: Frames in the video. The second row: Synthesized results without the expression detail mapping. The third row: Synthesized results with the expression detail mapping. The fourth row: Close-up shots of the synthesized faces without texture from different views.

Hence, when the facial animation data recorded from a live subject's performance can be reused, plenty of animator's work can be reduced.

For our future work, alignments of faces are made according to the positions of the corresponding features. In fact, the more markers placed on the face, the more accurate of the alignments. However, the more markers placed on a face, the more likely the details will be hidden by the markers themselves. This is the trade-off. As a result, only limited markers are placed on subjects' faces and the results of marker tracking are perturbed by measurement noise, therefore the alignment results are not very precise. As a result, some noise may appear in the ratio images and so is the result rendering.

Currently only the frontal view in the video is utilized to generate ratio images. If ratio images can be obtained by stitching the frontal and mirrored views into one and utilize a BSSRDF (Bidirectional Scattering Surface Reflectance Distribution Function)[16] to represent 3D face models, the resulting animation can be better.

References

[1] Lin I-C, Yeh J-S, Ouhyoung M. Extracting 3D facial animation parameters from multiview video clips. *IEEE Computer Graphics and Applications*, Nov/Dec. 2002, 22(6): 72–80.

[2] Liu Z, Shan Y, Zhang Z. Expressive expression mapping with ratio images. In *Proc. SIGGRAPH'01*, Los Angeles, CA, USA, 2001, pp.271–276.

[3] Platt S M. A structural model of the human face [Dissertation]. University of Pennsylvania, 1985.

[4] Waters K. A muscle model for animating three-dimensional facial expression. *Computer Graphics* (*SIGGRAPH Proceedings*), 1987, 22: 17–24.

[5] Williams L. Performance-driven facial animation. In *Proc. SIGGRAPH'90*, Dallas, Texas, USA, Aug. 1990, pp.235–242.

[6] Guenter B, Grimn C, Wood D. Making faces. In *Proc. SIGGRAPH'98*, Orlando, Florida, USA, Aug. 1998, pp.55–66.

[7] Bregier C, Covell M, Slaney M. Video rewrite: Driven visual speech with audio. In *Proc. SIGGRAPH'97*, Los Angeles, CA, USA, 1997, pp.353–360.

[8] Cosatto E, Graf H P. Photo-realistic talking-heads from image samples. *IEEE Trans. Multimedia*, 2000, 2(3): 152–162.

[9] Ezzat T, Geiger G, Poggio T. Trainable videorealistic speech animation. In *Proc. SIGGRAPH'02*, San Antonio, Texas, USA, 2002, pp.388–398.

[10] Wu Y, Kalra P, Moccozet L, Magnenat-Thalmann N. Simulating wrinkles and skin aging. *The Visual Computer*, 1999, 15(4): 183–198.

[11] Tiddeman B, Burt M, Perret D. Prototyping and transforming facial textures for perception research. *IEEE Trans. Computer Graphics and Applications*, Sep/Oct 2001, 21(5): 42–50.

[12] Bando Y, Kuratate T, Nishita T. A simple method for modeling wrinkles on human skin. In *PacificGraphics2002 Proceeding*, 2002, pp.166–175.

[13] Gonzalez R C, Woods R E. Digital Image Processing. Addison-Wesley Press, ISBN: 0-201-60078-1, 1992.

[14] Noh J-Y, Neumann U. Expression cloning. In *Proc. SIGGRAPH'01*, Los Angeles, CA, USA, 2001, pp.277–288.

[15] Lin I-C, Yeh J-S, Ouhyoung M. Realistic 3D facial animation parameters from mirror-reflected multi-view video. In *Proc. Computer Animation 2001*, IEEE Computer Society, Nov. 2001, pp.2–11.

[16] Jensen H W, Marschner S R, Levoy M, Hanrahan P. A practical model for subsurface light transport. In *Proc. SIGGRAPH'01*, Los Angeles, CA, USA, 2001, pp.511–518.

Pei-Hsuan Tu is currently a software engineer at CyberLink Corporation. She received the B.S. degree in computer science from "National Chung-Cheng University" in 2001, and M.S. degree in computer science from "National Taiwan University" in 2003. Her research interests include computer graphics and image processing. She is a member of IEEE and IEEE Computer Society.

I-Chen Lin received the B.S. and Ph.D. degrees in computer science from "National Taiwan University". His research interests include computer graphics, computer animation and motion tracking. He is a member of ACM SIGGRAPH, IEEE, and IEEE Computer Society.

Jeng-Sheng Yeh received a B.S. degree in computer science from "National Taiwan University" and is currently a Ph.D. candidate in "National Taiwan University". His research interests include computer graphics, computer user interface, and 3D protein retrieval. He is a member of ACM SIGGRAPH.

Rung-Huei Liang is a post doctoral researcher at the Communication and Multimedia Laboratory at "National Taiwan University". His research interests include facial/gesture recognition and virtual reality applications. He received the B.S. and Ph.D. degrees in computer science from "National Taiwan University".

Ming Ouhyoung is a professor of Dept. Computer Science and Information Engineering at "National Taiwan University". His research interests include computer graphics, virtual reality, and multimedia systems. He received the B.S. and M.S. degrees in electrical engineering from "National Taiwan University" and a Ph.D. degree in computer science from the University of North Carolina at Chapel Hill. He is a member of IEEE and ACM.