Personalized Facial Animation Based on 3D Model Fitting from Two Orthogonal Face Images

Yonglin Li and Jianhua Tao

National Lab of Pattern and Recognition, Institute of Automation,
Chinese Academic of Sciences, Beijing 100080
{ylli, jhtao} @nlpr.ia.ac.cn

Abstract. In this paper, a personalized MPEG-4 compliant facial animation system in embedded platform is presented. We report a semi-automatic and rapid approach for personalized modeling from two orthogonal face images. The approach is very easy and efficient. With multi-dimension texture mapping, the personalized face model offers a much more lifelike behavior. The system can be used in game, interactive services etc.

1 Introduction

Facial animation has already proved to be impressive in interactive services, like human-machine interactions or virtual reality application or web services. For example, such systems can be used in news reading, help desk, e-cogent, playmail etc [3]. Although the cartoon like characters can play expressive impression, they can not very attractive. If lifelike and personalized characters can be used, the visual quality will become much better.

Facial animation research started in the early 70's by Parke's [11] pioneer work. In the last decade the quality of facial animations has improved remarkably due to the development of hardware and corresponding software. But in these days the generating of lifelike animated faces still remains an open issue. Human were train since birth to recognize faces and scrutinize facial expression. So we are very sensitive to the slightest changes of the synthesized face model [3].

There have been many work focus on creating a personalized face model. One technology which has been used widely is from the 3D rang data [4, 17, 18]. Another technology is from two orthogonal images, one frontal view and one side view [5, 7, 8, 9].Others have make attempt from images or video sequences which contains multiple views of a person [19, 20, 21].

During the animation, many researchers have focused on personalized facial animation from two orthogonal face images, one frontal view and one side view, in this process; they usually extract the 3D position of the facial feather points from the two orthogonal images automatically or by additional manual work. A generic face model is then fitted according the corresponding 3D position of the facial feather points to make global adaption, local deformation are used to make other feather points. After the deformation, texture mapping were applied to ensure the reality.

Another technique is introduced by Takaaki Akimoto [5] etc. In his paper, the 3D generic model is projected according to the orthogonal images after the facial feather

J. Tao, T. Tan, and R.W. Picard (Eds.): ACII 2005, LNCS 3784, pp. 996 – 1003, 2005.

points are extracted. Interpolation method was then used to calculate the displacement of non-feather vertices. Texture mapping is also used from two joint face images. However, he didn't find the transformation during the deformation of the generic model to ensure the accuracy.

Liu [2] has improved the technique by using only one frontal view of the face image by adapting all the computing in 2D space, no 3D transformation or rotation is considered. It makes the process easier and more efficient, however, the frontal view image can only afford limit degree in 3D space during animation, no depth information is considered for the reality of personalized face model.

Unlike liu's work, we expand the method to two orthogonal images, the approach doesn't need to compute the 3D position of the facial feather points. And the transformation matrix of the generic model is computed to ensure the accuracy in deformation. This make the process easier and more efficient since all the computing were restricted in 2D space.

The animation process was made according to MPEG-4 facial animation framework [22, 23]. Currently, MPEG-4 defines facial animation technique depending on the displacement of the facial definition points (FDP) described by facial animation parameters (FAP). For very low bit-rate communication, MPEG-4 defines Synthetic-Natural Hybrid Coding (SNHC), which can achieve the 1kbps condition by using the 68 significant parameters including 2 high level parameters define the viseme and expression and 66 low level parameters define the motion behaviors.

Animation in the embedded platform have attracted more and more researcher [6,10,12], Thomas Di Giacomo[6] had presented an approach that automatically refines or simplifies 3D facial animation, including its transformation to 2D graphics, to be displayed at interactive frame rates on heterogeneous devices. It lucked reality in appearance and was not personalized in behavior since their model is only a generic model with proper material. Our system can show a personalized and lifelike face model in such platform which makes it very attractive in interactive services.

The paper is organized as follows: Section 2 presents our approach in detail. In section 3, we use the model in real application with embedded environment. Section 4, then, makes some discussing of the current and future works.

2 Personalized 3D Face Model Creation

Our approach begins with pre-processing of two orthogonal images, and then the generic face model is projected according to the two view images. After projection, model fitting algorithm is presented for automatic face adaption, while all feather points were fitted, Radial Basis Function (RBF) is used to the transformation result. To ensure the reality, multi-direction texture mapping technique is used. The procedure for personalizing a 3D face model is described next in details.

2.1 Pre-processing

See the fact that the head height of same person should be equal, suppose R= HF/HS, where HF is height of the head in the frontal view image, HS is height of the head in the side view image, the side view image can be scaled through R to make the two images exactly the same in head height.

Then facial feather points have to be extracted for the fitting process. In this step we extract the facial feather points semi-automatically, the eye, mouth corner and nose tip are extracted automatically, manual efforts were used to locate other feather points which can describe the face contour. The feather points were chose mainly based on their importance in the face images. In total 32 feature points in front view image and 20 feature points in side view image are selected.

2.2 3D Model Fitting

The generic model used is based on the IST facial model [24] which is widely used in the SNHC of MPEG-4. The back part of the face model is excluded to improve efficiency in the late rendering process. Fig. 1 shows the generic model from different view.

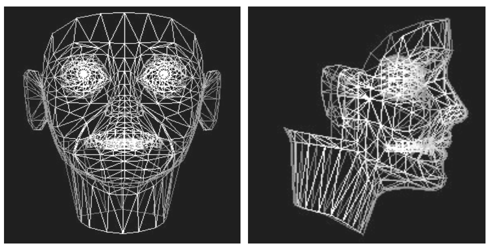

Fig. 1. Generic face model without hair part in different views

The adaptation process is as follows:

1) We project the generic model to XY plan and YZ plan according to the frontal and side view images, and select 32 and 20 corresponding points according to the face images in the generic face model.
2) Then, the 32 corresponding feather points in the XY projection were used. We suppose $P_{Mf} = (X_{Mf}, Y_{Mf})^T$ is the position of the feather points of the model, where $_{Mf}$ denotes the XY projection of the 3D model, $P_{If} = (X_{If}, Y_{If})^T$ is the position of the feather points of the face image, where $_{If}$ denotes the front view face image. s is the scaling factor. R is a $2*2$ rotation matrix, T is the translation vector. Eq.1 gives the sum squared error between the calculated points $\{P_{Mf1}....P_{Mfk}....P_{Mfn}\}$ $(k=1....32)$ and the corresponding feather points $\{P_{If1}....P_{Ifk}....P_{Ifn}\}$ $(k=1....32)$.

$$\text{Min } E(s,R,T) = \sum_{\text{points}} (P_{mf} - P_{if}) \tag{1}$$

Where

$$s \cdot R \cdot P_{mf} + T = P_{if} \tag{2}$$

With above results, procrustes [14, 15] analysis were used to compute the value of s, R, T iteratively. In the experiment, like Liu [2]'s result, we also find one iteration is sufficient in most cases.

3) After the 32 corresponding feather points were fitted, Radial Basis Function (RBF) was applied to deform other feather points of the generic model. Details can be found in [16].

4) We use $P_{Ms} = (X_{Ms}, Y_{Ms})^T$ to represent the position of the feature points of the generic model, where $_{MS}$ is the YZ projection of the 3D model, and use $P_{Is} = (X_{Is}, Y_{Is})^T$ to be the position of the feather points of the face image, where $_{Is}$ is the side view face image. Apply the algorithm in step 2 and 3 to fit the side face model. Fig. 2 shows the fitting result. From the experiment result we can see the approach performs very well.

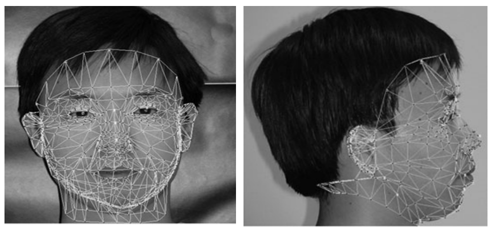

Fig. 2. Fitting result in front view and side view image

5) After that, the personalized face model can be directly reconstructed by combining the above two fitting results in both front view and side view. Let's define a 2D position ($X=X_f, Y=Y_f$) in the model from the front view and ($Z=Z_s$, $Y=Y_s$) from the side view. Then, the position (X, Y, Z) of the 3D face model could be got by:

$$\begin{cases} X=X_f; \\ Y=(Y_f+Y_s)/2; \\ Z=Z_s; \end{cases} \quad (3)$$

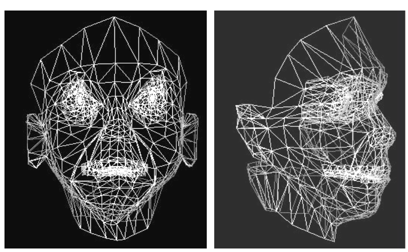

Fig. 3. Personalized model without texture in different views

During the fitting processing, we found that the Y_f and Y_s is quit similar. This partly proved that the algorithm play very well. Fig. 3. shows reconstruction results of personalized 3D face model.

6) Finally, the texture mapping are then applied to the model to ensure the reality, in the experiment multi-direction mapping algorithm were used by simply estimate the angle between the normal vector of the triangles in the model and the Z axis, to determine whether the frontal image or the side image should used. To join the front and side view images smoothly, the system maps a blended image where the texture mapping encounter each other, the blending ratio is according to the location of the personalized face model. This makes the texture mapping process more simple in computation. The result can be seen as follows.

Fig. 4. Personalized face model with texture mapping

3 Real-Time Animation in Embedded Platform

Animation comes directly when the personalized face model has created. But it faces difficulty when the animating process display on embedded platform like mobile phones and personal digital assistant with limit CPU and RAM. So proper computer graphic rendering technology has to be considered.

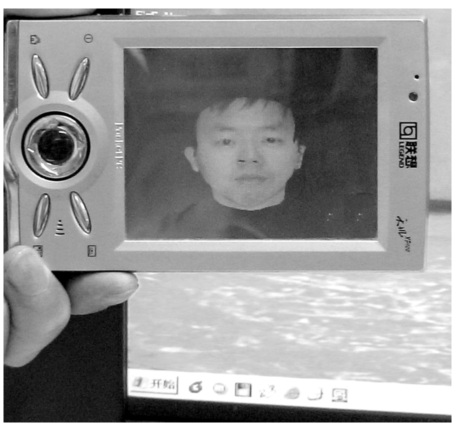

Fig. 5. Personalized facial animation in embedded platform

The animation processing was tested in a personal digital assistant with 206-MHz processor and 64Mbytes main memory. In fact, only 6-8MB was available when the animation process enabled. In the experiment, the mesh vertices were simplified to 780 and 1640 triangle in total; this is suitable for many embedded platforms. The floating-point arithmetic was converted to fixed-point arithmetic also. And Klimt [25] is also used to replay standard OpenGL function. These improvements can make the animation and rendering real-time. As expected. The personalized face model can displayed an expressive result. It can display 10--20 Fps (Frame rates per second) which is enough for real time animation in such platform. Fig. 5 and 6 have showed the animating result.

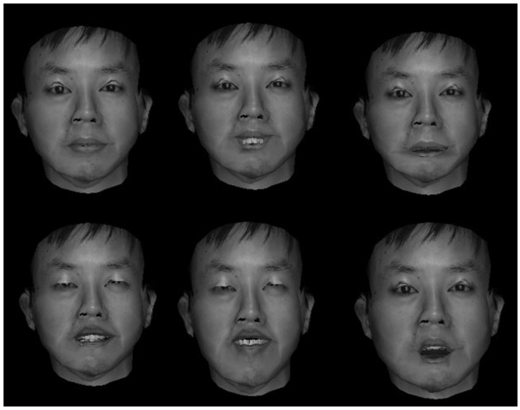

Fig. 6. Generating different expression for the personalized face model

4 Discussion and Future Work

Compared to the original techniques, the system is more efficient and easy, the computation is only applied in two dimension space, and this is different from the original techniques where the computation made in three dimension spaces while fitting the feather points. So our approach shows better quality in computation complexity. This makes the modeling more quickly.

Unlike just interpolating to face model projection, the system use mean square errors regulation and procrustes analysis [14, 15] to compute the fitting matrix. This has ensured the precision during the matching process. These improvements had enhanced the reality of personalized face model compared to original techniques which can be found in Fig.4 and 6. The improvements in computation complexity and reality in personalized face model are showed in Tab.1.

Table 1. Compares with other techniques

Methods / Characteristics	A-Nasser Ansari[9]	Takaaki [5]	Our approach
Computation complexity	high	Low	Low
Reality	Low	Low	High

The system has proved to be real-time in the embedded platform successfully. So it might be able to be used in the application 3G platforms in the future. Further improvements should be focus on the fully automatic extraction of facial feature points and proper computer graphic techniques for enhancing the rending efficiency in embedded platform.

5 Conclusion

In this paper, a semi-automatic and rapid approach for creating a personalized face model from two orthogonal face images is presented. The approach is very easy and efficient compared to original approaches. We also report a real-time MPEG-4 compliant facial animation system in embedded platform. With multi-dimension texture mapping, the personalized face model offers a much more lifelike behavior.

References

1. Jianhua Tao, Tieniu Tan. Emotional Chinese talkinghead system. International Conference on Multimedia Interface,2004
2. Zicheng Liu. A fully automatic system to model faces from a single image. Technical report MSR-TR-2003-55
3. Eric Cosatto, Jorn Ostermann, Hans Peter Graf, Juergen Schroeter. Lifelike Talking faces for interactive services. Proceeding of IEEE special issue on multimedia human computer interaction. September,2003
4. Yuencheng Lee, Demetri Terzopoulos, Keith Waters. Realistic modeling for facial animation. Proceedings of SIGGRAPH 1995
5. Takaaki Akimoto, Yasuhito Suenaga, Richard S.Wallace. Automatic creation of 3D Facial Models. IEEE Computer Graphics& Applications. 1993
6. Thomas Di Giacomo, Hyung-Seok Kim, Stephane Garchery, Nadia Magnenat-Thalmann. Benchmark-Driven automatic transmoding of 3D to 2D talking heads. Workshop on Modelling and Motion Capture Techniques for Virtual Environments, December 2004
7. Nikos Grammalidis, Nikos Sarris, Michael G.Strintzis. Generation of Personalized MPEG-4 compliant talking heads. International Workshop on Digital Communications (IWDC 2002), pp. 185-190, 2002.
8. Shiguang Shan, Wen Gao, Jie Yan, Hongming Zhang, Xilin Chen. Individual 3D face synthesis based on orthogonal photos and speech-driven facial animation. International Conference on Image Processing 2000
9. A-Nasser Ansari and Mohamed Abdel-Mottaleb. 3-D face modeling using two views and a generic face model with application to 3-D face recognition. Proceeding of the IEEE conference on Advanced Video and Signal Based Surveillance. 2003
10. Igor S. Pandzic. Facial animation framework for the web and mobile platforms. ACM Special Interest Group on Computer Graphics and Interactive Techniques 2002
11. F.I. Parke. Computer generated animation of faces. ACM National Conference. ACM, November 1972.
12. Florent Duguet, George Drettakis. Flexible Point-based rendering on mobile devices. IEEE Computer Graphics and Applications number 4 volume 24 July-August 2004
13. Murat Tekalp, Jorn Ostermann. MPEG-4 Facial Animation Project. http://www.cs.technion.ac.il/~gip/projects/s2002/Dmitry_Vadim_MPEG4/index.htm
14. J.R. Hurley and R.B.Cattell. The Procrustes program: Producing a direct rotation to test an hypothesized factor structure Behavior Science.1962，pp258–262
15. AL Yuille, PW Hallinan, DS Cohen. Feature extraction from faces using deformable templates. International Journal of Computer Vision, 1992
16. Floater, M.S., and A. Iske, Multistep Scattered Data Interpolation using Compactly Supported Radial Basis Functions, Journal of Computational and Applied Mathematics vol.73, no.5, pp. 65-78, 1996

17. L.S.Chen, T.S.Huang, J.Ostermann. Animated Talkinghead with Personalized 3D Head Model. IEEE first Workshop on Multimedia Signal Processing,1997.
18. V.Blanz and T.Vetter. A Morphable Model for the Synthesis of 3D faces. In Proc. SIGGRAPH99,1999.
19. Frederic Pighin, Richad Szeliski, David H.Salesin. Modeling and Animating Realistic Faces from Images. International Journal of Computer Vision 50(2),143-169,2002.
20. Z.Liu, Z.Zhang, G.Jacobs and M.Cohen. Rapid modeling of animated faces from video. Journal of Visualization and Computer Animation, 12(4):227-240,Sep,2001.
21. M Dimitrijevic, S Ilic, P Fua. Accurate Face Models from Uncalibrated and Ill-Lit Video Sequences. Conference on Computer Vision and Pattern Recognition 2004
22. ISO/IEC IS 14496-2 Viusal, 1999
23. A.Murat Tekalp, Jorn Ostermann. Face and 2-D mesh animation in MPEG-4. Signal Processing: Image Communication, 15 (2000) 387~421
24. Gabriel Antunes Abrantes, Fernando Pereira. MPEG-4 Facial Animation Technology: Survey, Implementation, and Result. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 9,NO.2,March,1999
25. Klimt- the open source 3D Graphics Library for Mobile Devices http://studierstube.org/klimt/