Expressive Textures

K. Fei

University of Pretoria, Computer Science Department, Pretoria 0002, South Africa
kkfei@hotmail.com

Abstract

If a moving image is more expressive than words or than a still image, then an animated facial expression can explain more in depth the feelings of a virtual character. Facial animation has been used in many applications, from entertainment to research on virtual humans and tele-presence. The aim of most of the approaches is to achieve high degrees of realism of virtual characters and is supplemented by complex models of kinematics, muscle movement, movement of clothing as well as cognition and behavioral models. Video avatars and image-based techniques are also used for creating virtual humans. However, the complexity of the geometric and physically simulated facial models used by the above methods make them unsuitable for use in distributed collaborative virtual environments running on low bandwidth networks or over the internet. Therefore, the majority of approaches for such environments are using simplified models of virtual human, which the obvious disadvantage of lower degrees of realism.
The Reflective Textures method, presented in this paper, makes use of textures of images of both synthetic faces or faces captured from video and of a simple low-polygon face/head model. It provides an interactive way of fine tuning and adjusting the underlined model to allow a more realistic mapping for a specific facial image. Furthermore, it concentrates in creating facial expressions by manipulation of the texture. Facial Expressions such as Fear, Happiness, Melancholy or Surprise, blinking of the eyes and movement of the pupils is automatically achieved for any mapped facial image by the system.

Key words: facial expressions, collaborative virtual environments, character animation, texture mapping

1 Introduction

Human faces are one of the most complex objects to model in computer graphics and especially in real-time. There are many attempts to truly model facial expressions some as old as computer graphics itself.

State of the art research in virtual humans models the face mesh and simulates movement of the skin, layered muscles and lower jaw bone that result highly realistic Facial expressions. To achieve these high level of realism the approaches demand high computing power and often specialized computer graphics hardware.

Other approaches use texture-mapping techniques, which can result in simulating facial expressions with a lower level of realism but with much improved performance and low computing and networking requirements.

A similar approach is presented in this paper. The approach is aiming at providing a fair amount of realism but keeping the model as simple as possible. The goal is to provide an approach suitable for low cost tele-presence, teleconferencing and distributed virtual environment applications. A low complexity head model is used together with texture mapping techniques to simulate a variety of facial expressions. The Reflective Textures method, uses textures from images of both synthetic faces or faces captured from video. It provides an interactive way of fine-tuning and adjusting the underlined model to allow a more realistic and accurate mapping of a specific facial image. Furthermore, it concentrates in creating facial expressions by manipulation of the texture. Facial Expressions such as Fear, Happiness, Melancholy or Surprise and features such as blinking of the eyes and movement of the pupils are automatically achieved for any mapped facial image by the system.

This paper first presents an overview of related research in synthetic humans with main focus in facial expressions. Section 3 describes in detail the expressive textures approach while section 4 gives examples of facial expressions and animations achieved with this method. Finally, discussion and conclusions on possible applications and results are given in section 5.

2 Related Work

The term *Synthetic Avatar* refers to a computer-generated entity or character. Virtual Human, which is another term for human-like synthetic avatars, are extensively used in virtual environments and for a variety of applications. For these characters to be effective and believable they need to be rendered in real time and perform actions without direct user control and with high degree of realism [13][14][15]. They often need to interact with the user who is exploring the Virtual Environment or, in distributed virtual environments, represent the user's embodiment in the virtual world. In these cases, interaction with and reaction to the user presence or response needs to be included.

Synthetic actors take many roles in today's virtual environments, including teacher [23], guide, companion [22], assistant[7], entertainer[7], and opponent [4][20][21].

Modeling of facial expressions for synthetic humans adds realism and achieves high levels of engagement and responsiveness. However, the task of realistically model all human expressions of a human on to a virtual character, is complicated by the richness of human facial expressions and the fact that each individual has their unique way of expressing emotions facially.

Approaches that achieve high degree of realism manipulate the face mesh's vertices but are complex and require large memory and computation power[1]. Face mesh manipulation approaches can be further divide into low-level muscle motion simulator, known as action units, abstract muscle action procedures [8], or minimum perceptible actions.

Face mesh manipulation of facial animation approaches includes *Keyframing*, which is one of the earliest approaches and involves linear transformations from one face mesh to another[25]. This approach requires extensive computation and large data sets. It is inflexible as the number of expressions that can be generated is restricted by the keyframes already digitized and it is also difficult to generalize from one face mesh to another.

Parametric Deformations, which model the human face as parametric surface and record the transformations as control points' movements in order to minimize the data storage requirements [5]. The limitation of this approach is that it is difficult to generalize over different face meshes.

One attempt is to utilize B-spline patches that are manually defined on an actual digitized face mesh and use the control points of the B-spline patch to move and distort the face mesh. This method is powerful, but there is no automatic way of defining the relevant control points for the B-spline patches [11].

The other attempt is to use rational free-form deformations to move points inside a defined volume with respect to the control points placed at the edges of the volume. The volume can be distorted by changing the position of the control points [19]. This approach has the same limitation as the above.

Anatomically correct muscles, this approach is to simulate the actual human face muscles. This approach can simulate large number of facial expressions, but this approach is complex and difficult to implement [3][18].

Pseudo-muscles, this hybrid approach simulates muscles that can be anatomically incorrect. It only takes into account the muscles that are related to facial animation and it is easier than the previous face mesh approaches and requires smaller data set [8].

Another category of approaches uses image manipulation, which is less computationally intensive but lacks the high degrees of realism. Image manipulation of facial animation is done by morphing the normal image of a neutral face that is pasted on a model of a head or a face mesh (moving pixels in the image). It can also be done by texture coordinates displacement, which changes the image's coordinates to simulate facial expression [16].

Expressive Textures is a texture coordinate displacement approach. It takes advantage of the low computation costs and to create a generic method for facial expressions that can be used over low bandwidth networks for distributed collaborative virtual environments. It uses one low complexity generic face mesh, onto which different faces are mapped. These faces can be synthetic images of video images of real faces. The method allows for interactive fine-tuning and adjustment of the specific facial image for better realistic mapping of a specific face, avoiding the need for creating more specialized face meshes.

3 Expressive Textures

The approach is divided in three steps, namely texture mapping and texture adjustment, facial expression creation and eye animation.

3.1 Texture Mapping

In a texture in OpenGL for example, a texel lies in the texture coordinate system and is between 0,1. Any mapping value larger than 1 is tiling (repeating) the texture.

Let $\lambda(x, y)$ be a pixel in the image and x and y are the coordinates of the pixel in terms of image size. Let $\mu(x, y)$ be the size of the image with x and y as the width and height of the image.

If m(x, y) is the texture mapping position require for the face mesh, we can express it as an equation:

$$mx = \lambda x/\mu x \quad , \qquad my = \lambda y/\mu y$$

which can also be expressed as:

$$m(x, y) = \lambda(x, y)/ \mu(x, y)$$

Many image applications can help us resize an image, according to a ratio, so that face images can fit into the generic face mesh. Unfortunately, we cannot simply resize the image by comparing the image size of the image that fit the mesh to the one we want to map. We need to ensure that for example the image is not from a shoulder height and that it includes only the face.

In Expressive Textures texture adjustment is achieved by first finding the distance between the center of the iris in both eyes in the facial image that correctly maps on to the mesh. Then divide that to the distance between the center of the iris in both eyes in the facial image we want to map onto the mesh.

It can also be done, by drawing a perpendicular line between the eyes and the tip of the nose. Use this line as distance for comparing the two images. Divide this distance in the image use in the face mesh to the image that requires mapping on the face mesh.

These results will give us the ratio we require scaling the image, and then we can cut away portions that are not required e.g. shoulders.

3.2 Facial Expressions

Simple Facial expressions are build up from simple movements of the facial parts namely: eyes, eyebrows, and the corners of the mouth.

Psychological research has classified six facial expressions that relate to distinct universal emotions: disgust, melancholy, fear, happiness, anger and surprise [3][6]. Notice that four of them are negative emotions: disgust, despair, fear, and anger. Only happy and surprise are classified as positive emotions.

Firstly we generalize the motion cues for each facial expression, as summarized in the following Table.

Expression	Motion cues
Melancholy	Lowering the mouth corners, raise the inner portions of the eyebrows.
Surprise	Eyebrows slide up, eyes wide open.
Fear	Inner eyebrows raise, eyes open, and mouth lowers slightly.
Happy	Raising mouth corners, lower outer eyebrows, eyes close slightly and upper cheeks expand slightly.
Disgust	Upper lip raised eyes close slightly and lower eyebrows.
Anger	Inner eyebrows lowered, outer eyebrows raise and lips press against each other or lower slightly.
Cunning	Eye lids nearly close the eyes, and mouth corners raise slightly

3.3 Eye Animation

In order to simulate real eye movements, we must find a technique to simulate eye rotations even when we are using still images. This is more complex compared to the approach that uses a face mesh with eye sockets and eyes balls, since simulating real eye movements is only a matter of rotating the two eyeballs. In the Expressive Textures masking is applied (using Transparency channel) so that a hole can be created on the image with the image of the pupils appear under this hole (see Figure 1).

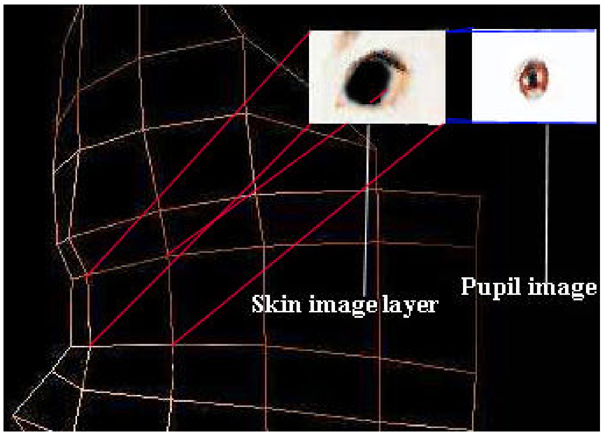

Figure 1. Shows the image of the skin layer with a hole created by masking appear over the pupil image

By moving the image of the pupils we can simulate eye movements (gaze direction).

Rotation of the Eyes

Not all animations require deforming the face mesh, for instance moving the eyebrows can be achieved by sliding the texture image over the wireframe.

With masking we can simulate the rotation of the eyes, since we can move the texture coordinate to achieve the same effect to rotation. The movement is calculated by:

Let a = eye image 's X or Y coordinate

Let b = new X or Y coordinate

Let ob = the starting X or Y coordinate

Let c = any small value decrease the amount of movement eg.0.0001

a = a - c*(b - ob);

When animating a facial expression the eyeballs should be able to rotate freely same as real eyes, because the facial expressions are independent from the eyeballs rotation.

With masking the underlying pupil image is independent from the image of the skin layer, when animating only the skin layer is morphed according to the motion cues define previously. During animation of facial expression the pupil image can rotate in any direction.

4 Design and Implementation

The Expressive Textures are implemented in OpenGL, using C++ as and the images are in bitmap formats.

4.1 Generation of the Face Mesh

The first part of the implementation is to manually create a face mesh that contains fewer polygons, while looking fairly realistic. The face mesh consists of 138 vertices and 238 polygons, and was generated without any scanner or equipment that can digitize the face mesh from a visual input. (see Figure 2). The face mesh remains generic. If the generic face mesh is modified to the actual face of the user, it will require more processing for extraction of facial features and face model modification from visual input [17][24].

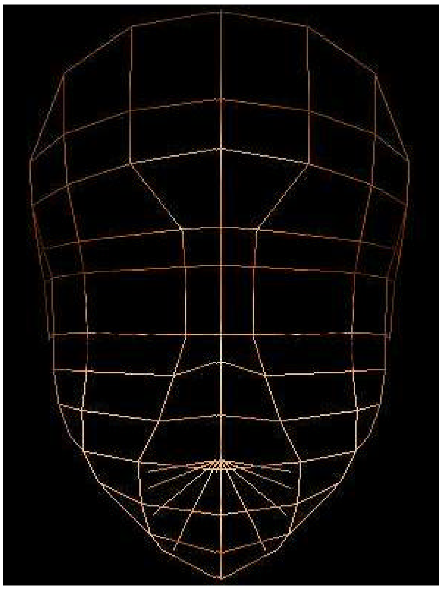

Figure 2. The face mesh after generation

The face mesh is divided into stripes of polygons, which divide the face into regions, so that texture mapping can be done easier. It can be adjusted for different facial images by moving the texture coordinates of each strip, as described by the formula of section 3.1 (see Figure 3).

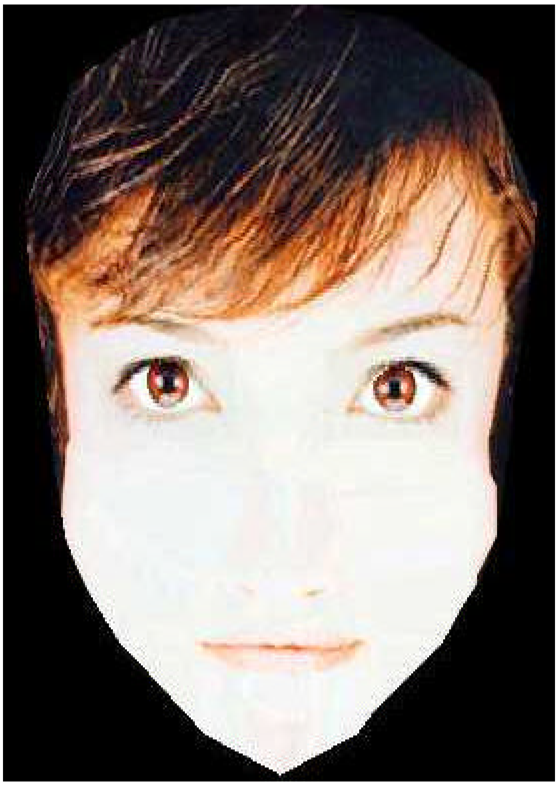

Figure 3. The face mesh after texture mapping with a facial image

4.2 Facial Expressions

Once the face mesh is correctly textured we can start animating the expressions by first finding the texture coordinates for each expressions motion cues.

Then classifying them according to which facial parts they animate, for example grouping all movement associated with left eye together. The reason for this classification is because we can pick from previously defined facial cue’s movements and creating new, different facial expressions. The second reason is that it makes implementation easier to track or change any facial parts movement without affecting other parts. Therefore, there are grouped into different arrays.

By moving the texture coordinates that map the facial parts at a neutral position to the texture coordinates that associate with an expression the face is animated to that facial expression. For instance, to simulate the eyes closed we move the two upper textures coordinates closer to the two lower texture coordinates that map the eyelid texture. (see Figure 4)

Figure 4. a) The eyes are at neutral position. b) The two upper texture coordinates move closer to the two lower texture coordinates that map the eyelid texture to simulate eye close**.**

But to simulate eyes completely closed, we will require another still image of the same face with the eyes closed, so that we can merge the two images to simulate eyes blinking or eyes closed (see Figure 5)

Figure 5. The face image's right eye is merge with the image of the face with close eyes.

In addition, from the facial cues that we have already defined in the theoretical approach, we notice that we will need to morph the mesh when simulating a smile. This will improve the realism of the model, which cannot be achieved by texture coordinates displacement alone.

With the texture coordinates for each motion cues, we can now simulate the six universal expressions as well as other expressions by combining the motion cues. (see Figure 6). Notice that happy expression also expands the mesh's vertices that are lying at the cheek region.

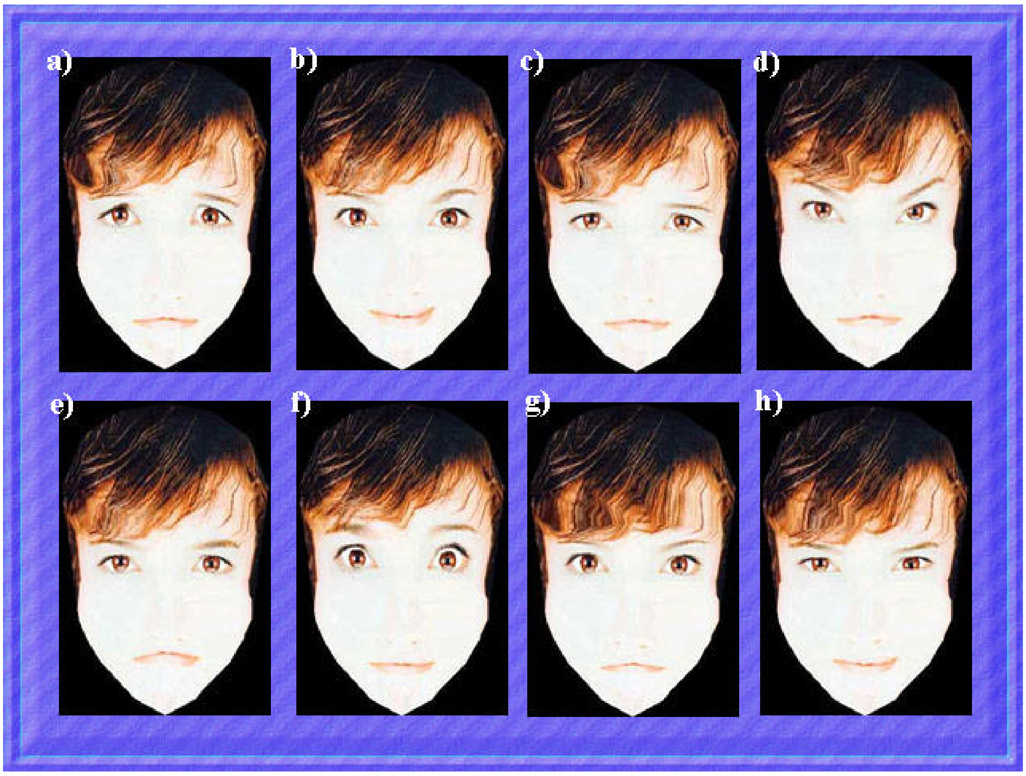

Figure 6. The six universal expression and other expressions. a)Fear b)Happy c) Melancholy d)Angry e)Disgust f)Surprise g)Teasing h)Cunning

4.3 Eye Animation

In implementation, we can create holes in the skin layer image by first providing a mask, which indicates the portion in the image which is going to be transparent via the Alpha channel (transparent channel). One separate texture of the pupil behind the skin texture can be displaced to alter the face's gaze direction (see Figure 7)

Because of the alpha channel, the shape of the eye contours remains unchanged and covers the texture portion.

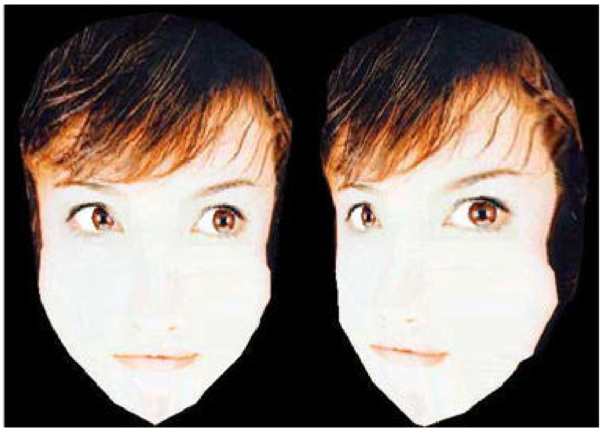

Figure 7. Shifting of the texture of the pupils to the left, giving the result after 3D mapping

The pupil image is generated by applying masking with only the pupils visible in the original face image, which lies under the skin layer. By displacing the mask and the image, eye movements are simulated.

Even when the Facial Expressions are been animated, the eyes can still rotate under the skin layer, because there are two separate texture layer.

In order to make the application dynamic, we assume that in some situations we cannot supply the masks and the transition face image (face with eyes close), and provide both a masking and a non-masking mode.

In masking mode, the mask of the facial image and image is provide and we can simulate eye movements and facial expressions from it. In non-masking mode, only the facial image is given and we can only simulate the facial expressions, but the pupils will be slightly distorted (see Figure 8)

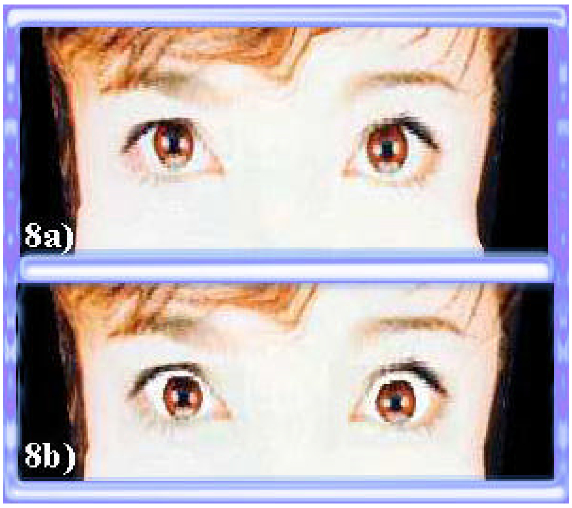

Figure 8. Shows the different between masking and non-masking mode. In 8a) The pupils are distorted, because masking mode is off, while in 8b) the pupils are not distorted as masking is on.

Finally, Figure 9 gives an overview of the user iterface of the Expresive Textures tool.

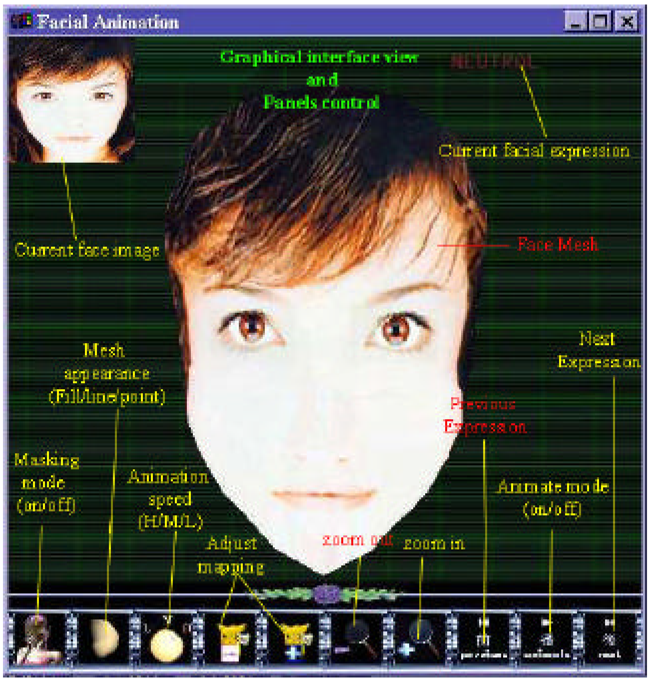

Figure 9. User Interface

5 Conclusions

The result of using texture coordinates displacement in simulating facial expression is fairly realistic, and if an image with skin folds (wrinkles) are available the Expressive Textures approach can use blending to achieve this effect, while keeping the computations at the lowest and using fewer system resources.

When still images are used the trade off is that this approach can distort the hair on the image, if the person's hair is lying close to the eyes or eyebrows. Another limitation is that mapping an open mouth onto a closed mouth is not realistic, if one tries to model human speech with this approach.

Expressive Textures is a useful approach for numerous applications that require use over a network, like a virtual chat system, as well as in collaborative distributed virtual environments and low bandwidth teleconferencing.

If combined with body motion it can be also used in entertainment, to create expressive animated characters that show facial expressions/emotions together with body motions. This for example can improve the computer games where virtual actors have only have body expressions.

6 References

[1] Brooke, N.M. & Scott, S.D., *Computer graphics animations of talking faces based on stochastic models*, Proceedings of the International Symposium on Speech, Image-processing and Neural Networks, Hong Kong, 1994, IEEE, pp. 73-76.

[2] G.Sannier, N.Magnenat-Thalmann, "A User-Friendly Texture-Fitting Methodology for Virtual Humans", Computer Graphics International'97, 1997.

[3] K. Waters, "A Muscle Model for Animating Three-dimension Facial Expression", In SIGGRAPH'87, volume 21 of Computer Graphics Annual Conference series, July, 1987, pp.17-24.

[4] L. Emering et al., *Interacting with Virtual Humans through Body Actions*, IEEE Computer Graphics and Applications, vol. 18, no. 1, pages 8-11, 1998.

[5] M. Escher, N. Magnenat-Thalmann, "Automatic 3D Cloning and Real-Time Animation of a Human Face", Proc. Computer Animation, IEEE Computer Society, pp. 58-66, 1997.

[6] M.J. Black, Y. Yacoob, Tracking and Recognizing Rigid and Non-Rigid Facial Motions using Local Parametric Models of Image Motions, ICCV 1995

[7] N. Jennings, and M. Wooldridge (1996) Software Agents. IEEE Review, January 1996, pp. 17-20.

[8] N. Magnenat-Thalmann, E. Primeau, and D. Thalmann, Abstract Muscle Action Procedures for Human Face Animation, Visual Computer, 1988, 3, pp.290-297.

[9] N.Magnenat-Thalmann, D.Thalmann, "Animating Virtual Actors in Real Environments", ACMMS'97, Switzerland, Springer, Vol.5, No2, 1997, pp.113-125.

[10] N.Magnenat-Thalmann, I.S.Pandzic, J-C.Moussaly, "The Making of the Terra-Cotta Xian Soldiers", Digital97, Geneva, Switzerland, Digital Creativity, 8(2), July 1997, pp.66-73.

[11] Nahas M., Huitric H., Rioux M., and Domey J., Facial Image Synthesis Using Skin Texture Recording, Visual Computer, December 1990

[12] Parke, F.I Techniques for facial animation. In N. Magnenat-Thalmann and D. Thalmann (Eds.) New Trends in Animation and Visualization. John Wiley, Chichester, 1991, P.229-241

[13] P. Maes et al. The ALIVE system: Full-body interaction with Autonomous Agents, Proceedings of the Computer Animation '95 Conference, Geneva, Switzerland, IEEE-Press, April 1995.

[14] P. Karp and Steven Feiner. Issues in the automated generation of animated presentations. In Proceedings of Graphics Interface '90, pages 39-48, May 1990.

[15] P. Karp and Steven Feiner. Automated presentation planning of animation using task decomposition with heuristic reasoning. In Proceedings of Graphics Interface '93, pages 118-127, Toronto, Ontario, Canada, May 1993. Canadian Information Processing Society.

[16] S. Valente, J. Dugelay, Face Tracking and Realistic Animations for Telecommunicant Clones, Eurecom Institute, IEEE MultiMedia P.34 - 43 (January - March 2000)

[17] W.S Lee, P.Kalra, N.Magnenat-Thalmann, "Model Based Face Reconstruction for Animation", Proc. MMM'97 (World Scientific Press), Singapore, 1997, pp.323-338.

[18] Y. Lee, D. Terzopoulos, K. Waters, Realistic modeling for facial animation, In Computer Graphics (Proc. SIGGRAPH), pp. 55-62, 1996.

[19] P. Kalra, A. Mangili, N. Magnenat Thalmann, D. Thalmann, "Simulation of Facial Muscle Actions Based on Rational Free Form Deformations", Computer Graphics Forum (Proceedings of Eurographics'92), Blackwell Publishers, Oxford, 1992, pp.59-69.

[20] M. Moshell, "Virtual Environment in the US Military", Computer vol. 26, February 1993, pages 81-82.

[21] T. W. Mastaglio and R. Callahan, "A Large-Scale Complex Virtual Environment for Team Training", Computer vol. 28 no7, July 1995, pages 49-56.

[22] A. E. Johnson, M. Roussos, J. Leigh, C. R. Barnes, C. A. Vasilakis, and T. G. Moher, “The Nice Project: Learning Together in a Virtual World”, Proc. VRAIS 98, IEEE Computer Society Press, Los Alamitos, California, 1998, pages 176-183.

[23] T. Noma, L. Zhao, N. I. Badler, “Design of a Virtual Human Presenter”, IEEE Computer Graphics and Applications, July/August 2000, pages 79-85.

[24] Takaaki Akimoto, Yasuhito Suenaga and R. Wallace, “Automatic Creation of 3D Facial Models”, IEEE Computer Graphics & Application, September, 1993, pp.16-22.

[25] F.I. Parke and K. Waters, ``Computer Facial Animation," A K Peters, 1996.