Journal of VLSI Signal Processing 20, 97–105 (1998)

Animated Talking Head with Personalized 3D Head Model

JÖRN OSTERMANN
AT&T Labs—Research, Room 3-231, 100 Schultz Dr., Red Bank, NJ, 07701

LAWRENCE S. CHEN AND THOMAS S. HUANG
Beckman Institute CSL, University of Urbana, IL 61801

Abstract. Natural Human-Computer Interface requires integration of realistic audio and visual information for perception and display. An example of such an interface is an animated talking head displayed on the computer screen in the form of a human-like computer agent. This system converts text to acoustic speech with synchronized animation of mouth movements. The talking head is based on a generic 3D human head model, but to improve realism, natural looking personalized models are necessary. In this paper, we report a semi-automatic method for adapting a generic head model to 3D range data of a human head obtained from a 3D-laser range scanner. This personalized model is incorporated into the talking head system. With texture mapping, the personalized model offers a more natural and realistic look than the generic model. The model created with the proposed method compares favorable to generic models.

1. Introduction

Human-Computer Interface is an application area where audio, text, graphics and video are integrated to convey various types of information. Often conversion is necessary between different media [1]. The objective is to provide more natural interaction between the human user and computer. One approach is to display an animated character or a life-like talking head on the computer screen, with the ability to receive input from the user and respond in a natural and intelligent way. Such an agent may perform information retrieval, reading email or replying to email messages. Already available are software programs that display a generic animated talking head or a cartoon animal character on the screen to perform various tasks. One program is able to fetch songs at the user's request [2]. Another is able to carry on simple conversations [3]. Yet others are able to convert written text into visual speech using a Text-To-Speech (TTS) synthesizer and a synchronized talking head with realistic lip and jaw movements, and with visible teeth [4] and tongue [5, 6].

Most programs use a generic model that is deformable according to a set of parameters [7]. However, to make such interaction more resembling to that of a human-to-human interaction, realistic and natural looking animations are required [8]. This calls for models that are based on real measurements of the structures of the human face, as well as facial features such as color, shape and size. Such information can be computed using image analysis techniques [9–11]. In this contribution, we use information gathered using a 3D-laser scanner that produces very dense range data of the human head. In addition, it also provides the corresponding color image of the head. This information can be utilized to achieve a more naturally looking talking head than a generic model.

This paper describes results of fitting a generic head model to a set of 3D range and color data. The fitted model is incorporated into an existing Visual Text-To-Speech (VTTS) system, and the color image is used for texture mapping for animation.

In Section 2, we describe the architecture of our VTTS system. In Section 3, the creation of personalized 3D head models from range data is described. Results and a comparison are presented in Section 4.

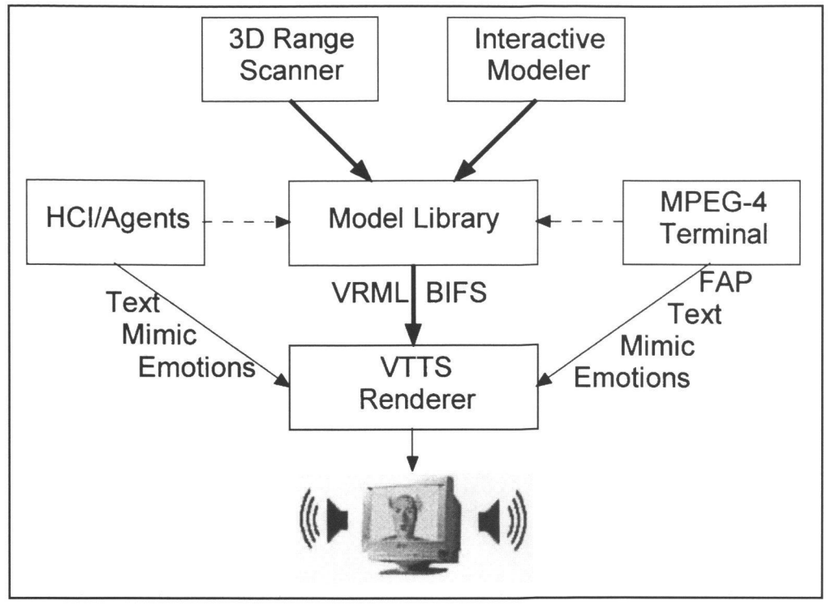


Figure 1. Concept of a Visual Text-To-Speech (VTTS) synthesizer with an animated talking head system.

2. Visual Text-to-Speech System

The major application areas of talking head systems are in human-computer interfaces, in customer service as well as in games where persons want to control artificial characters. For both applications, we want to be able to make use of easy to animate and easy to modify characters. Figure 1 shows the block diagram of a versatile VTTS system that allows animating different characters, human like or avatars.

As can be seen from the diagram, a model library provides several models that can be read by the VTTS system. The application can be the human-computer interface, an agent or an MPEG-4 communications terminal. The application selects a model from the library and sends it to the VTTS renderer. Depending on the application, this model is described in VRML format [12] or in BIFS format [13] as defined by MPEG-4. After loading the model, the renderer can receive animation data from the application. Input is text and facial animation parameters like facial expressions or the Face Animation Parameters (FAP) as defined by MPEG-4 [14]. The renderer computes the image of the face model and synthesizes the text that it receives from the application.

In our implementation of the talking head system, we use the AT&T text-to-speech (TTS) program FlexTalk. The program converts text to synthetic speech. The text is parsed and analyzed extensively to produce the speech. The phonemes and the related timing information are used as input of the visual animation system for animating mouth movements. The animation subsystem uses a parameterized 3D model, which is a descendant of Parke's model [7], further improved with a coarticulation model for synthetic visual speech developed at UC Santa Cruz [15]. The structure of the 3D model is a wireframe of numbered vertices in 3D coordinate space, with connections specified to form polygons (Fig. 2). Prescribed colors are added to each polygon to form smooth-shaded surfaces (Fig. 3). This sophisticated model includes the face, eyes, mouth, teeth and tongue, and is capable of producing very realistic mouth movements. It is controlled by a set of deformation parameters. With a set of time-varying parameters, an animation sequence is produced. Occasional head movements and eye blinking are added heuristically.

This system gives satisfactory performance with synchronized speech and lip movements. However, this talking head (Fig. 3) may seem impersonal because it does not represent any person the human user may know. If one can produce personalized models, the interface would improve, and the user may even have the choice of selecting from models of several different persons.

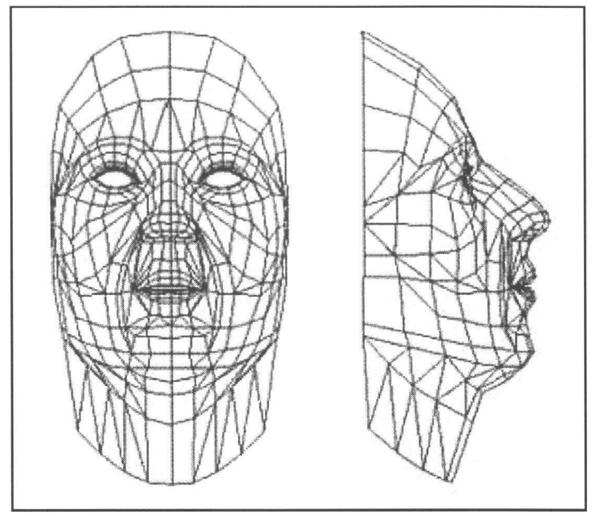


Figure 2. Wireframe of the generic model.

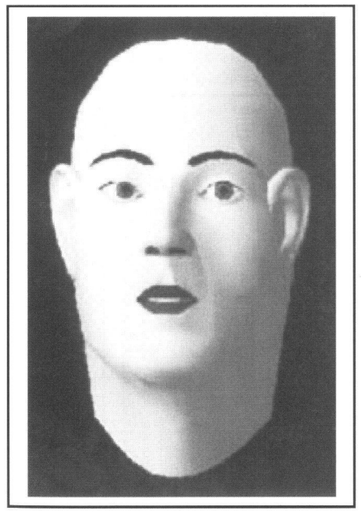


Figure 3. Smooth-shaded generic model.

In general, we can imagine two ways of creating face models. One would be to create a model by hand using modeling software. A second approach is considered in this paper. Here, we focus on the goal to fit the generic model (Fig. 2) to 3D range data of a person's head.

3. Personalized 3D Head Model Creation

As input for our system, we use the data from a 3D-laser range scanner. This data is then used in order to adapt the generic face model (Fig. 2) to the individual person represented by the range data.

3.1. 3D Range Data

The 3D scanner used is the Cyberware 3D laser range scanner. The subject sits in a chair while the scanner revolves around the person to scan the surface structure of the head. The scanner gives a very dense set (over 260,000 points) of range data in cylindrical coordinates. In addition to the range data, a color camera captures the surface color of the head, so that for each range point the corresponding color is also available. This is useful for texture mapping in the final rendering of the talking head. Figure 4 shows two rendered views of the 3D range data. Figure 5 shows the texture map of the scanned person.

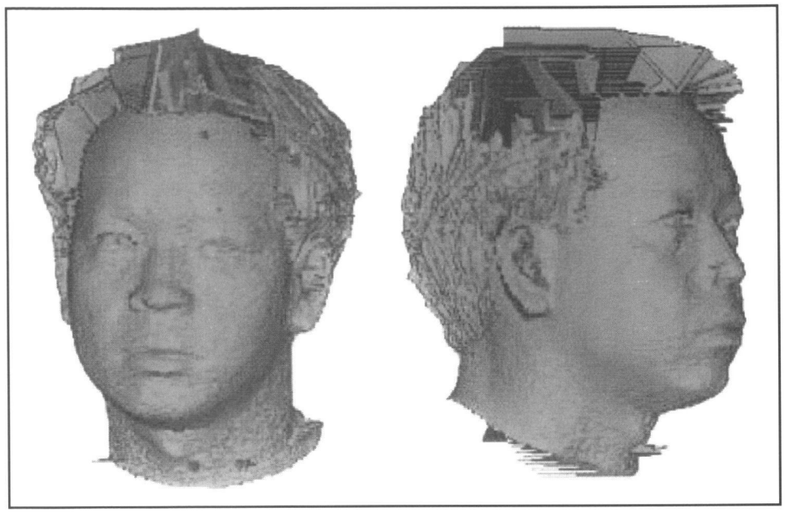


Figure 4. Two views of the 3D range data.

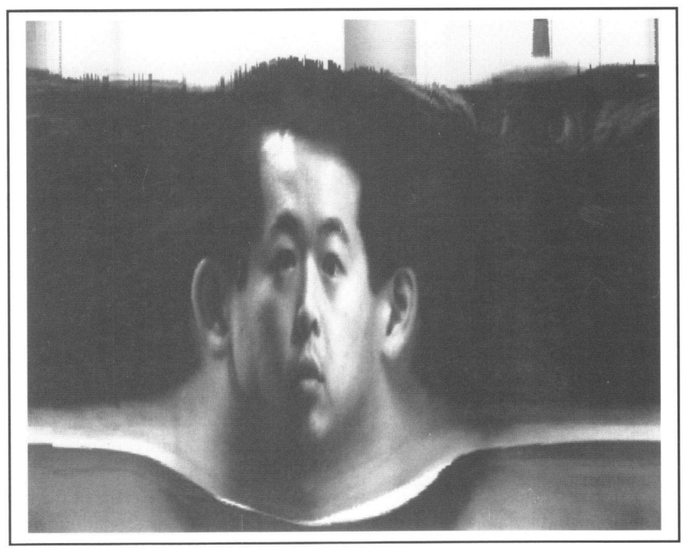


Figure 5. Texture map of the 3D range data shown in Fig. 4.

3.2. Model Fitting

In this section, we describe fitting of the generic model to the 3D range data. Initially, the generic head model is larger in scale than the range data. Vertical scaling factors are first obtained to scale down the model, and the vertical profile line along the center of the face is fitted. After the profile line is fitted, the rest of the face is fitted through radial projection.

The range scanner gives a data set in the left-handed cylindrical coordinate system (r_L, y_L, ϕ_L) while the wireframe model is in the usual right-handed Cartesian coordinate system (x, y, z). They are transformed into the same right-handed cylindrical coordinate system (r_R, y_R, ϕ_R) by

$$r_R = r_L, \quad y_R = y_L, \quad \phi_R = 2\pi - \phi_L \tag{1}$$

and

$$r_R = \sqrt{x^2 + z^2}, \quad y_R = y, \quad \phi_R = \arctan\frac{x}{z} \tag{2}$$

To find the vertical profile line, we first find the tip of the nose. The tip of the nose is generally the most protruded structure of the face, i.e., its distance is greatest to an imaginary vertical rotation axis in the center of the head. This point is selected manually, but it can also be detected automatically. After the point is obtained, the profile line is determined (Fig. 6). In case that the profile line is not aligned with the vertical axis of the range data, the profile line is determined by an additional point along the vertical line, in our case a manually marked point on the center of the chin. From the profile line, feature points such as the indent above the nose, the upper lip, mouth center and the lower lip are detected automatically based on the curvature of the profile line (Fig. 7). Neighboring feature points are used to divide the range data into eight horizontal slices. Each slice corresponds to a slice defined by the same feature points of the generic face model. The location and separation distances of these features and the thickness of the slices are used to define vertical scaling factors. The algorithm is not sensitive to the exact vertical alignment as long as the vertical line crosses the areas where the feature points are located. This is because the curvature of the surface along the profile line does not change rapidly.

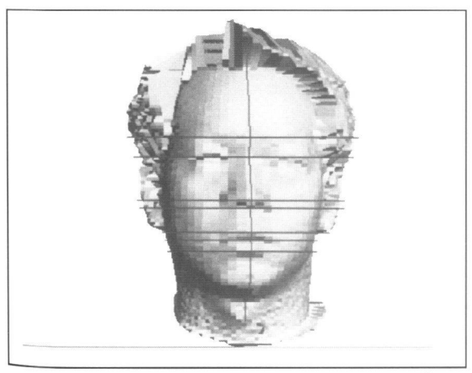


Figure 6. 3D range data with profile line and feature points marked on the profile line.

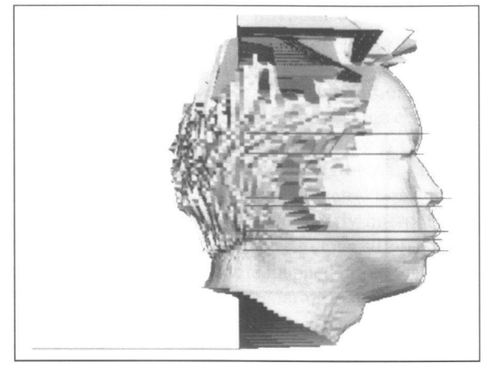


Figure 7. 3D range data with profile line and feature points marked on the profile line (side view of Fig. 6). The feature points divide the range data into eight horizontal slices.

Each scaling factor is used to scale down one slice of the generic model to the size of the range data in the vertical direction. Then each vertex point on the model is radially projected onto the surface of the range data. For each vertex point, we have the height y_R and the angle ϕ_R in the cylindrical coordinate system. Then we trace a ray radially back towards the vertical axis y, piercing the range data. This ray may not intercept the range data at exactly one range point, so the nearest four range data points are averaged to give the new coordinate for the model vertex. The radial projection is depicted in Fig. 8.

This fits the skin part of the generic model to the range data, but the eye and mouth positions are critical and need to be adjusted manually. Right now, only the face is fitted, but the procedure can be easily extended to fit the entire head including ears and the back of the head. The fitted model is then ready to be incorporated into the VTTS system.

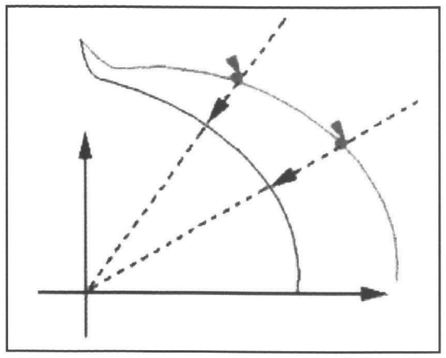


Figure 8. Radial projection: the inner curve is the range data, and the outer curve is the wireframe model. The places where the rays intercept the range data are the new coordinates for the model.

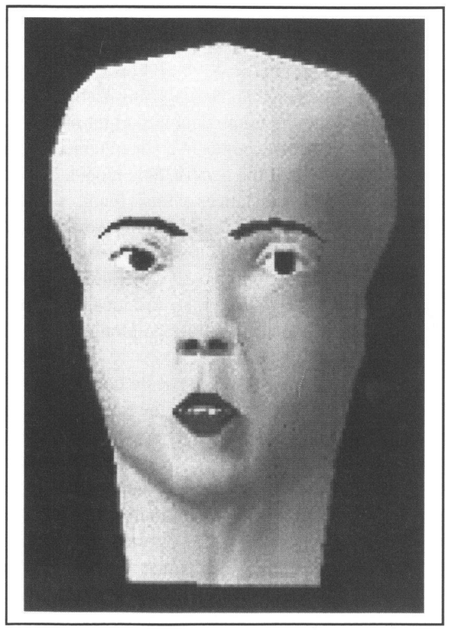


Figure 9. Fitted model with smooth shading.

4. Results

Figure 9 shows the talking head with the new model with smooth shading. At first glance, it does not appear to be much better than the original model (Fig. 3). Since the original model does not contain polygons to model

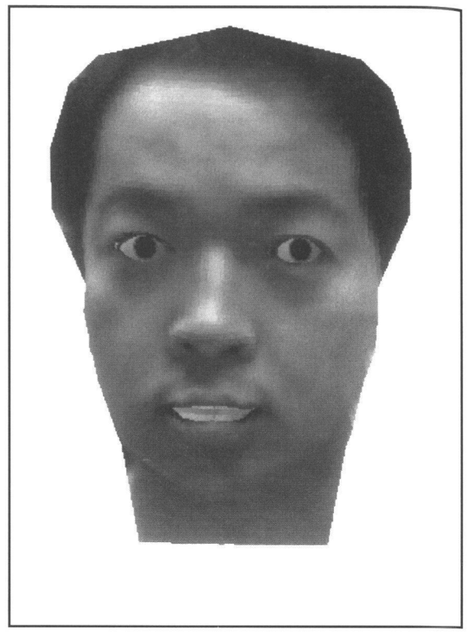


Figure 10. Fitted model with smile and texture mapping.

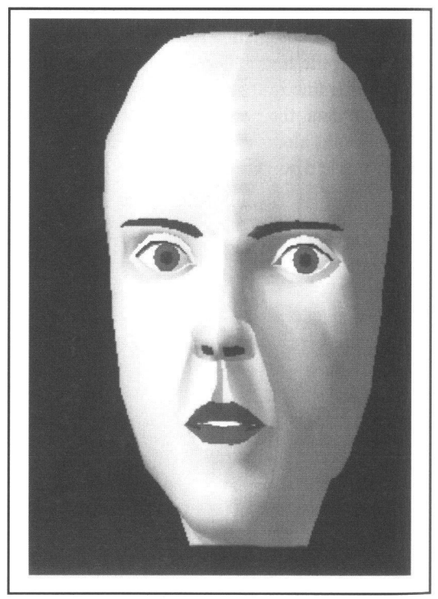


Figure 11. A second fitted model with smooth shading.

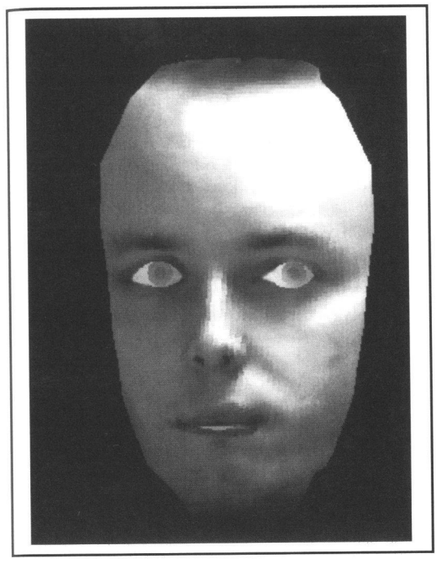


Figure 12. A second model with texture mapping.

the boundary between the forehead and the hair, it cannot be precisely adapted to the 3D range data providing significant detail in this region. This problem is overcome by texture mapping. Instead of smooth shaded polygons with prescribed colors, the color and texture of the polygon surface of the face come from a color image (Fig. 5). The image is literally pasted onto the polygonal wireframe. Geometric transformation is done automatically by bilinearly interpolating the texture through the computer graphics library routines. Figure 10 is the texture mapped rendering of the head. Clearly, the resulting rendering is much more realistic. Figures 11 and 12 show a second example of a face model created from a wire frame.

During animation, some limitations of the model become obvious. From the texture map (Fig. 5), it is not possible to extract separate texture maps for eyeballs, teeth or tongue. These are necessary in order to allow for realistic texture mapping when the eyelids are moving or the model is speaking. In order to allow for realistic animations, we have to use generic texture maps for these parts of the head (Fig. 10). Realistic texture maps can be taken from manually created models (Fig. 13). When comparing the automatically and

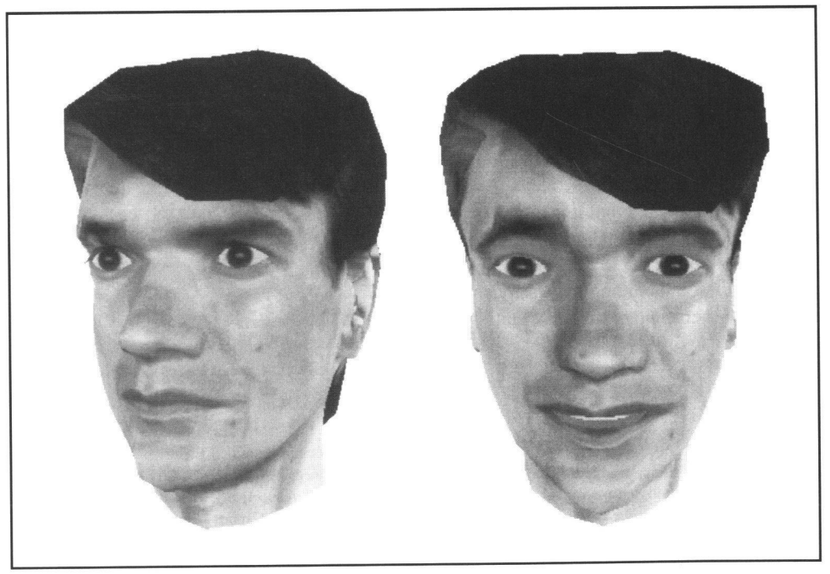


Figure 13. Model created manually from several photographs with realistic texture maps for eyeballs, teeth and tongue.

manually created models, the manually created model shows advantages (Fig. 13) in the precise modeling of the hairline and ears.

When comparing the shaded models (Figs. 9 and 11) with the textured models (Figs. 10 and 12), the limitation of the mouth adaptation is visible. Only mouth corners are adjusted to the range data. Hence, the width of the lips is not correctly adapted. However, this is only seen in the shaded models due to the lip color. It does not show up in the textured models.

5. Conclusions

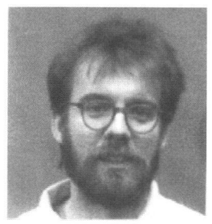
In this paper we describe the semi-automatic fitting of a personalized 3D model for a visual text-to-speech system to the 3D range data of a person. After the user identifies manually the tip of the nose in the range data, the algorithm extracts automatically several feature points of the face and uses them to adapt a generic face model to the range data. The fitted face model with a texture map of the modeled person looks more realistic and more believable than a simple generic model. In order to allow for good facial animation, we have to use generic texture maps for teeth, tongue and eyeballs. Alternatively, these texture maps can be created from separate pictures of the person to be modeled.

As far as informal subjective evaluation is concerned, the models are able to give the impression of a real person to a layperson for a limited time. They are not convincing to someone who knows the person that the model represents. Therefore, the model might be useful for customer service applications where the customer does not know the speaker but it is certainly not yet suitable to replace a conventional videoconference call.

We are now working on including expressions with the model such that the personalized model would smile occasionally [6].

References

1. S. Morishima and H. Harashima, "A media conversion from speech to facial image for intelligent man-machine interface," *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 4, pp. 594–600, 1991.
2. D. Kurlander and D.T. Ling, "Planning-based control of interface animation," Technical report MSR-TR-95-21, Microsoft Research, Redmond, WA, USA, Jan. 1995
3. K. Nagao and A. Takeuchi, "Speech dialogue with facial displays: Multimodal human-computer conversation," *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pp. 102–109, 1994.
4. G. Potamianos, E. Cosatto, H.P. Graf, and D.B. Roe, "Speaker independent audio-visual database for bimodal ASR," *Proceedings of the ESCA/ESCOP Workshop on Audio-Visual Speech Processing*, Rhodes, Greece, Sept. 1997.
5. R. Sproat and J. Olive, "An approach to text-to-speech synthesis," in *Speech Coding and Synthesis*, W.B. Kleijn and K.K. Paliwal (Eds.), Elsevier Science, 1995.
6. J. Ostermann and E. Haratsch, "An animation definition interface: Rapid design of MPEG-4 compliant animated faces and bodies," *International Workshop on Synthetic—Natural Hybrid Coding and Three-Dimensional Imaging*, Rhodes, Greece, pp. 216–219, Sept. 1997.
7. F.I. Parke, "Parameterized models for facial animation," *IEEE Computer Graphics and Applications*, Vol. 2, pp. 61–68, Nov. 1982.
8. Y. Lee, D. Terzopoulos, and K. Waters, "Realistic modeling for facial animation," *Computer Graphics, SIGGRAPH'95*, pp. 55–62, 1995.
9. M. Escher and N. Magnenat Thalmann, "Automatic 3D cloning and real-time animation of a human face," *Proc. Computer Animation'97*, 1997.
10. T. Akimoto and Y. Suenaga, "Automatic creation of 3D facial models," *IEEE Computer Graphics and Applications*, pp. 16–22, 1993.
11. Horace H.S.Ip and Lijun Yin, "Constructing 3D individualized head model from two orthogonal views," *The Visual Computer*, Springer-Verlag, Vol. 12, pp. 254–266.
12. J. Hartman and J. Wernecke, *The VRML Handbook*, Addison Wessley, 1996.
13. ISO/IEC JTC1/SC29/WG11 N1825 MPEG-4 Systems Working Draft V 5.0, Stockholm meeting, July 1997.
14. ISO/IEC JTC1/SC29/WG11 N1797 MPEG-4 Visual Working Draft V 4.0, Stockholm meeting, July 1997.
15. M.M. Cohen and D.W. Massaro, "Modeling coarticulation in synthetic visual speech," in *Computer Animation'93*, Tokyo, M. Thalmann and D. Thalmann (Eds.), Springer-Verlag, 1993.



Jörn Ostermann studied Electrical Engineering and Communications Engineering at the University of Hannover and Imperical College London, respectively. He received Dipl.-Ing. and Dr.-Ing. from the University of Hannover in 1988 and 1994, respectively. From 1988 till 1994, he worked as a Research Assistant at the Institut für Theoretische Nachrichtentechnik conducting research in low bit-rate and object-based analysis-synthesis video coding. In 1994 and 1995 he worked in the Visual Communications Research Department at Bell Labs. He has been working with Image Processing and Technology Research in AT&T Labs-Research since 1996.

From 1993 to 1994, he chaired the European COST 211 sim group coordinating research in low bit-rate video coding. Within MPEG-4, he organized the evaluation of video tools to start defining the standard. Currently, he chairs the Ad hoc Group on Coding of Arbitrarily shaped Objects in MPEG-4 Video.

His current research interests include video coding, shape coding, computer vision, 3D modeling, face animation, coarticulation of acoustic and visual speech., computer-human interfaces, speech synthesis.

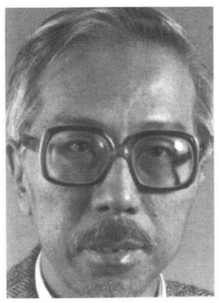
ostermann@research.att.com

Lawrence S. Chen received his B.S. degree and M.S. degree both in Electrical Engineering from University of Illinois at Urbana-Champaign. He is currently a Ph.D. candidate at University of Illinois and recipient of the Eastman Kodak fellowship.

He has been a research assistant at the Beckman Institute for Advanced Science and Technology at University of Illinois since 1992. In the summer of 1996 he worked at AT&T Labs—Research as a Member of the Technical Staff. In the summer of 1997 he spent three months at ATR Media Integration and Communications Labs in Kyoto, Japan, as an intern researcher.

His research interests include image processing, computer graphics, computer vision, and speech processing for applications in Human-Computer Interaction. His current project involves recognition of human emotions using multimodal information for natural human-computer interface.

lchen@ifp.uiuc.edu



Thomas S. Huang received his B.S. degree in Electrical Engineering from National Taiwan University, Taipei,Taiwan, China; and his M.S. and Sc.D. degrees in Electrical Engineering from the Massachusetts Institute of Technology, Cambridge, Massachusetts. He was with the Faculty of the Department of Electrical Engineering at MIT from 1963 to 1973; and with the Faculty of the School of Electrical Engineering and Director of its Laboratory for Information and Signal Processing at Purdue University from 1973 to 1980. In 1980, he joined the University of Illinois at Urbana-Champaign, where he is at present William L. Everitt Distinguished Professor of Electlrical and Computer Engineering, and Research Professor at the Coordinated Science Laboratory, and Head of the Image Formation and Processing Group at the Beckman Institute for Advanced Science and Technology.

During his sabbatical leaves: Dr. Huang has worked at the MIT Lincoln Laboratory, the IBM Thomas J. Watson Research Center, and the Rheinishes Landes Museum in Bonn, West Germany, and held visiting Professor positions at the Swiss Institutes of Technology in Zurich and Lausame, University of Hannover in West Gernmany, INRS-Telecommunications of the University of Quebec in Montreal, Canada and University of Tokyo, Japan, He has served as a consultant to numerous industrial firms and government agencies both in the U.S. and abroad.

Dr. Huang's professional interests lie in the broad area of information technology, especially the transmission and processing of multidimensional signals. He has published 12 books, and over 300 papers in Network Theory, Digital Filtering, Image Processing, and Computer Vision. He is a Fellow of the International Association of Pattern Recognition, IEEE, and the Optical Society of American; and has received a Guggenheim Fellowship, and A.V. Humboldt Foundation Senior U.S. Scientist Award, and a Fellowship from the Japan Association for the Promotion of Science. He received the IEEE Acoustics, Speech, and Signal Processing Society's Technical Achievement Award in 1987, and the Society Award in 1991. He is a Founding Editor of the International Journal Compute Vision, Graphics, and Image Processing; and Editor of the Springer Series in Information Sciences, published by Springer-Verlag.

huang@ifp.uiuc.edu