Face-to-Face Implies no Interface

Dennis Burford and Edwin Blake

CS01-15-00

Collaborative Visual Computing Laboratory
Department of Computer Science
University of Cape Town
Private Bag, RONDEBOSCH
7701 South Africa

e-mail: dburford@cs.uct.ac.za, edwin@cs.uct.ac.za

Abstract

It is important to design user interfaces that are intuitive and "natural" to use. This is especially true when computers mediate communication and interaction between people. Collaborative virtual environments (CVEs) encourage just this type of direct interaction. Because facial expressions are essential to face-to-face communication, we believe facial animation for avatars is important to virtual communications of this sort. We present a system for real-time expression tracking and facial animation using facial markers, low-cost cameras and PCs. We have tested our system with live video input and animated three different types of faces, including a 3-D muscle model.

1 Introduction

Computers are empowering - they enable people to perform complex tasks effectively and efficiently. Unfortunately, interacting with a computer is often a difficult and daunting chore, especially for the uninitiated. It is important, therefore, to design user interfaces that are intuitive and "natural" to use. This is especially true when computers mediate communication and interaction between people. Ideally, the intervening computers should "disappear" and leave a completely transparent communication experience.

Collaborative virtual environments (CVEs) encourage just this type of interaction. CVEs differ from other communications technologies (such as email, chat, telephony and video conferencing) in that they immerse the user in the medium. Users are encouraged to behave as they would in reality, instead of making use of a controlling interface. CVEs allow a deeper collaboration because participants share a single, common environment.

When using CVEs, it is important that the participants have a natural way of communicating and indicating their emotional state and attitude. Facial animation for avatars is one way of achieving this. Early work in manipulating facial expressions and gestures of avatars relied on an explicit control such as a panel of buttons or dials controlling predefined motions [7]. This approach was extended to use simple sketches of faces indicating basic expressions [30].

The underlying limitation of this technique is that users have to go through an interface in order to show their emotions. The resulting expressions can be neither spontaneous nor smooth. Also, if real-time interaction is needed, the user's experience is sure to be lessened because of the conscious effort required to drive the animation.

More recent work has attempted to animate the facial features of avatars directly from the expressions of the participants. These real-time performer-driven animation systems offer a more promising approach to conveying expressions.

In this paper, we present such a system for direct control of facial expressions for avatars through real-time facial expression tracking and synthesis. Our system uses a low cost "web-cam" and a set of markers for expression analysis. Once the markers have been placed on the face, very little control is required and the facial expressions are conveyed directly.

We have tested our system through the implementation a working prototype on the Windows platform. The prototype is able to track and synthesise facial expressions under changing lighting conditions. The system transmits facial expression parameters across a network and animates them remotely.

In the following section we give a background to the importance of faces to human-computer interaction. We also outline the major issues in facial animation, including models, parameterisation and performer-driven animation. Next we present our system for tracking facial features and quantifying expressions. After this we discuss our 2-D and 3-D animations and describe a prototype system we have developed. We end with an outline of future work and a conclusion.

2 Background

2.1 Faces and Computers

When people communicate face-to-face, facial expressions form an integral part of the interaction. Expressions provide a context for our speech by indicating both the nature and extent of our attention and meaning. In fact, it has been reported that facial expressions contribute 55% to the effect of a spoken message [23].

Furthermore, facial expressions have been shown to be universally expressed and recognised. Charles Darwin's pioneering work [9] gave credence to the fact that people around the world share the same expression of their emotions. The research of Ekman and colleagues [13, 11] affirmed these theories.

The consequences of these findings are significant for human-computer interaction. From the computer user's perspective, a face and its expressions are a form of "interface" that almost everyone will be able to relate to. For a computer system monitoring user behaviour, people will tend to show the same expressions when feeling the same underlying emotions , allowing for consistent interpretation of mood and emotion.

The utility is clear: people's innate face and expression recognition may be used to convey information and make interfaces more accessible [28]. This would be especially useful for users from diverse cultural backgrounds and for those generally unfamiliar with computers. For example, [31] have tried to make computer resources more accessible to the public by building information kiosks with human-like faces as the main interaction mechanism.

In addition to providing a natural interface, computers should be able to react more meaningfully to human users, a property that has been termed perceptual intelligence [26]. Pentland defines perceptual intelligence as "paying attention to people and the surrounding situation in the same way another person would, thus allowing these new devices to learn to adapt their behaviour to suit us, rather that adapting to them as we do today" [26]. Facial expression recognition and interpretation play a large role in this process.

2.2 Facial Animation

The focus of this paper is not on the recognition and analysis of human expressions, but rather the faithful conveyance of these expressions to other users, via the animation of computer models. The field of automated expression recognition, analysis and interpretation is rich and varied. For more information, we direct the reader to a review of expression recognition and classification systems by Pantic and Rothkrantz [24].

The importance of computer facial animation has not gone unnoticed. This is an active and diverse field of research. Applications include intelligent and autonomous agents; automated virtual news-readers; talking heads for information presentation [8]; entertainment, such as animated feature films, cartoons and games; virtual video-conferencing; and realistic avatars in virtual environments [17].

2.2.1 Face Models

The first major work in computer facial animation was done by Parke [25] who developed a 3-D parameterised face model. Platt et al [27] and Waters [32] created more complex models that simulated muscle movements. Lee et al [20] extended the concept of the muscle model to create a physically realistic model that more accurately simulated the multiple muscle and skin layers making up the human face.

The general trend has been towards physically realistic 3-D models. In contrast to this, recent work has sought to create non-photorealistic 2-D models [19, 4]. There has been evidence that simple 2-D models can accurately convey most of the basic expressions [28] and may even be more effective for virtual interactions [19, 4]. It would seem that users are often disappointed with human-like models that behave artificially. Their expectations are lower, or simply different, when cartoon models are used. Exaggerated or unnatural expressions are deemed acceptable [19].

2.2.2 Parameterisation Schemes

A number of schemes have been developed to describe and quantify facial expressions. The most significant of these is the Facial Action Coding System (FACS) developed by Ekman and Friesen [12]. FACS was developed to quantify facial movement in terms of muscle actions. It is a comprehensive method for classifying expressions and is used mainly for the analysis of human expressions by trained FACS experts. Magnenat-Thalman et al [22] developed an Abstract Muscle Action model (AMA) that was inspired by FACS, but designed for use in facial animation.

More recently, MPEG have defined facial animation and definition parameters as part of the MPEG-4 Synthetic Natural Hybrid Coding (SNHC) standard [1, 16]. These parameters can be used to describe the shape of a facial model and the movement of its facial features. The MPEG-4 scheme facilitates low bandwidth face-to-face communication by providing a standard for the transmission of expression parameters alone, instead of entire images as in standard video-conferencing.

2.2.3 Performer Driven Animation

The most convincing method for facial animation is often a performer driven approach that uses the expressions of the participant to drive a synthetic face. Williams [33] developed one of the first performer-driven facial animation systems by tracking face markers and deforming a facial texture map. More recently, Guenter et al [18] developed a sophisticated system for tracking facial markers and animating a 3-D mesh. Their system produced highly realistic (non-real-time) animations but required a large number of markers and a rig of six cameras.

A recent real-time performer-driven animation system is that of Goto et al [17]. They tracked facial features directly (without markers) and used the MPEG-4 facial animation parameters (FAPs) to quantify the expressions. A realistic 3-D model was animated using free form deformations. Another real-time system is that of Buck et al [4]. They too tracked features directly, but in contrast to [17] they animated non-photorealistic hand-drawn faces.

This type of animation is an active area of development with many commercial systems emerging to take advantage of the potential for low bandwidth real-time face-to-face communication. Many companies offer virtual cloning services [14, 3, 15, 10] to construct avatars from two orthogonal photographs of users. The avatars are then animated using speech or video input from the user.

Our system is most similar to the performer driven animation systems in this section. We drive animations of 2-D and 3-D characters in real-time by tracking the expressions of a user with the aid of facial markers. We use a low cost camera (a "web-cam") and a PC for tracking and animation. The tracking is discussed in more detail in the following section.

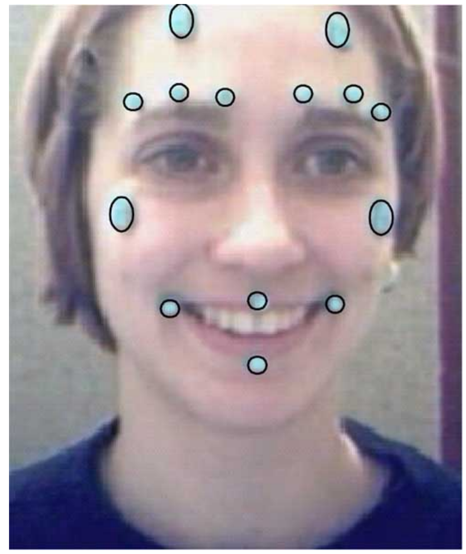

Figure 1: Layout of facial markers on the face.

3 TRACKING

We use 14 brightly coloured beads as markers to aid feature tracking. 4 markers are used to correct for global head motion (discussed later), another four are placed around the mouth and three are placed above each eyebrow. See Figure 1 for the layout of the markers on the face.

The markers simplify the recognition problem and ensure that the tracking is robust and fast, despite the use of a relatively low quality camera and non-constant lighting conditions. Systems have been developed to track facial features directly, without the use of markers. Few of these systems run in real-time or use low-cost cameras for video input, however. Also, many are unable to handle a diverse ethnic range of users. In the future, we hope to extend our tracking algorithm and reduce the dependence on markers.

In the following sub-sections, we discuss our tracking system in a "top-down" fashion, beginning with video input and ending with the low-level routines used for marker location.

3.1 Video Capture

An inexpensive "web-cam" is used for video input. We have chosen the Creative Web-Cam Go (USB), which provides video at 320x240 resolution at 30 fps. We use an ordinary PC running Windows to perform both the tracking and later animation.

3.2 Search Regions

The tracking system constructs and maintains search regions for the mouth, each eyebrow and for each of the 4 reference markers use to correct for global head motion. The search regions are kept slightly larger than the bounding box of the markers they are tracking. In this way, we can use coherence in marker positions to aid the tracking. Search regions are also constrained to ensure that they do not overlap.

After the search regions have been automatically detected in an initialisation procedure, only these areas of the image are scanned for markers. If there is a problem and a number of markers are lost, the system recovers by first locating the face in the image and then attempting to redefine the search region boundaries.

3.3 Sampling

Within each region, the image is sampled. We use an adaptive sampling procedure that increases the sampling rate until the required number of markers is found. The sampling is done in such a way that no pixel is revisited on each pass.

3.4 Segmentation

Each sample pixel taken in the search regions is tested to see if it is part of a marker. Standard image segmentation techniques have been employed to identify the markers. Region growing is performed for each sample that falls within predefined Hue-Saturation-Value (HSV) colour thresholds, and the centroid of each pixel cluster is calculated to give a single 2-D position for each potential marker. Pixel clusters are evaluated for acceptability based on their location and a number of size and shape criteria.

4 CORRECTING FOR GLOBAL HEAD MOTION

We do not constrain the user's head in any way, as we believe head movements form important gestures in face-to-face communication. Besides the obvious nodding and shaking, the user could hold her head to the side in an uninterested manner, or look down to show embarrassment or shame.

However, in order to accurately measure expressions it is important to correct for the effects this global head motion. We use 4 reference markers to aid in the correction process. In an initialisation process, the user is asked to look directly at the camera - the plane of the face being parallel to the camera image plane - and hold a neutral expression. The configuration of the markers, particularly the reference markers, is recorded during this time.

We can think of the reference markers as defining a quadrilateral in the image. Now, for every frame that the expressions are tracked the system determines the current reference quadrilateral. Using results from projective geometry [29] and image warping [34], we calculate the homography that transforms the current quad back to the initial reference quad. The homography is a plane projective transformation that describes the changes in position of projected points from a plane, after some rotation or translation of that plane.

Using this homography, we transform all the marker positions back to the "initialisation space". We are now in a position to measure the displacements of the markers from a neutral face and thereby describe the current expression. Figure 2 illustrates the correction process.

Of course, the above method makes the assumption that a face can be approximated by a plane. This leads to some error for large rotations. Also, the calculation of the homography is affected by noise in the tracking of the marker positions, leading to further errors. We apply a low-pass Gaussian smoothing filter in an attempt to reduce this noise.

4.1 Estimating Pose

The homography calculated in the previous step can be used in estimating the pose, or orientation, of the user's head. The matrix representing the homography can be decomposed using standard results from projective geometry [21]. Orientation parameters may then be extracted that describe the relative pitch, yaw, roll and translation of the head. These parameters completely define the head's pose and can be used in later animations.

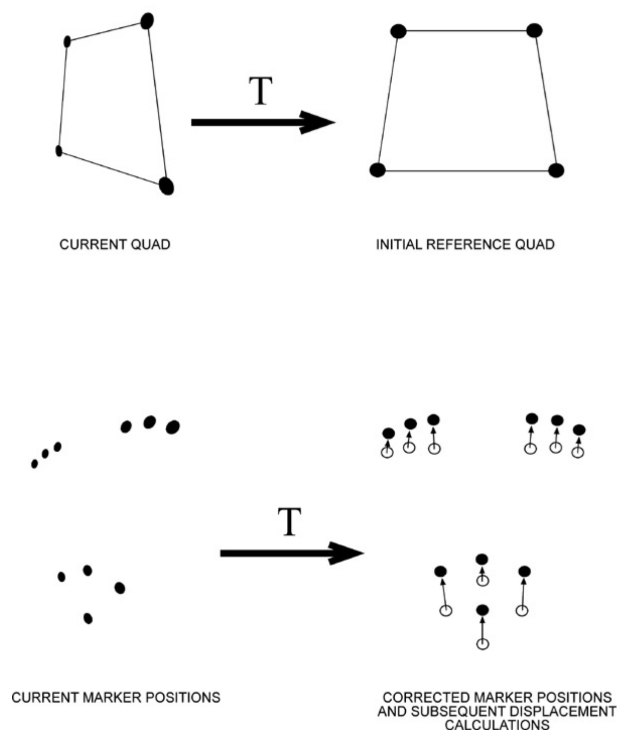

Figure 2: Correcting for the effects of rigid motion.

5 NORMALISING EXPRESSIONS

It is desirable to be able to animate faces of different shapes and characteristics. In order to do this we must take into account the properties that make each face unique, such as the width of the brows and the distance between the eyes. These properties are collectively called the face's conformation. The conformation of the user's face needs to be accounted for when measuring (and then synthesising) expressions. We use an approximation of the user's conformation to "normalise"' the measured expression displacements. We estimate the maximum possible displacements for the various features and use these values to scale the measured displacements. This method bears some similarity to the MPEG-4 FAP scheme.

6 ANIMATIONS

We have developed three different animations to convey the tracked expressions to remote users. The first two animations are 2-D techniques using a cartoon face and a more realistic facial texture. The third technique uses a 3-D model developed by Waters [32].

6.1 2-D Animations

The first animation is a simple 2-D cartoon face made up of a number of polygons representing features such as the mouth, eyes, cheeks and eyebrows. The polygons are layered from back to front and move in response to the normalised animation parameters received from the tracking system. The cartoon face is initiated to have the same conformation as the user and this conformation is used to interpret the normalized parameters. Several of the cartoon face's features are dependent on a single animation parameter. For example, the marker on the bottom lip of the user controls the cartoon's mouth, jaw and chin. This animation bears some similarity to the more complex cartoon-like animations of the CharToon system [19]. In fact, our tracking system could be used to animate their models in a similar way. Figure 3 shows an example face from our cartoon animation.

Our second technique attempts to create a more realistic 2-D animation by mapping images of actual faces or portraits onto a grid of connected nodes. The animation parameters are used to warp these facial textures in response to changing expressions. The warps attempt to simulate the facial feature movements in the image.

We have used the well know morphing algorithm of Beier and Neely [2] to warp our textures.

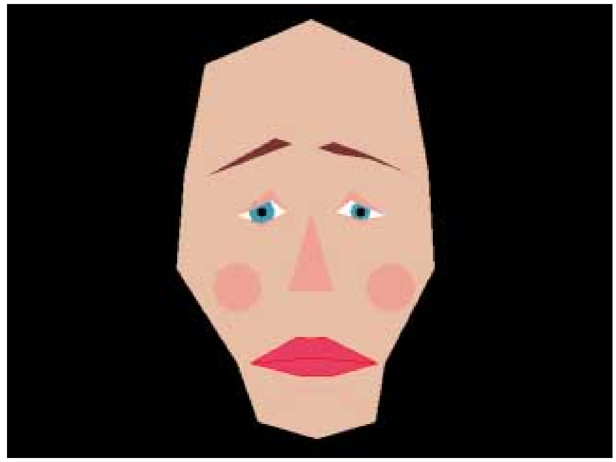

Figure 3: Cartoon face with a sad expression.

Their algorithm uses a series of morph-lines that are placed along corresponding features in source and target images. The endpoints of these lines are interpolated from source to target and the surrounding pixels are warped accordingly.

In order to ensure real-time performance, we have modified this technique slightly. The original algorithm warped every pixel in the image, whereas we warp only the nodes of our grid. Also, the original technique considers every morph-line for every pixel of the image. Instead, we define an area of influence around each morph-line. Only those nodes falling within this area are affected by that line. Figure 4 shows a screenshot from our tool for defining morph lines and conformation points.

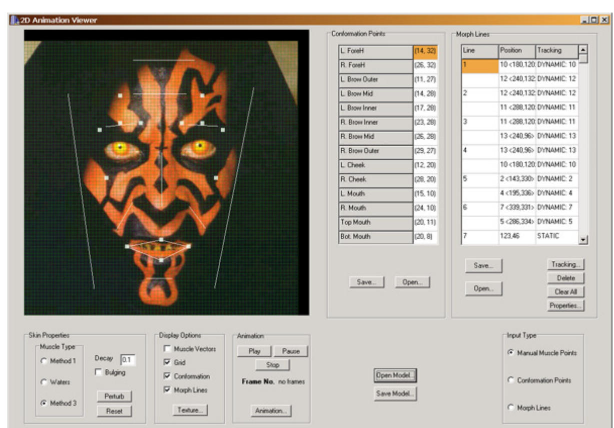

Figure 4: Tool for defining morph-lines and conformation points for the Beier-Neely algorithm.

6.2 3-D Animation

Our final animation uses a 3-D model of a face. The model is defined by a relatively sparse mesh of vertices and triangles. We use a modified version of Waters' technique to simulate muscle contractions and animate the mesh. In this section we briefly describe Waters' muscle model and then discuss our adaptations for use with tracked data.

6.2.1 Waters' Muscle Model

Waters' muscle model was developed to simulate muscle contraction and the skin's elastic response. It approximates facial muscles by several 3-D muscle vectors that placed in the appropriate positions around the facial mesh. See Figure 5.

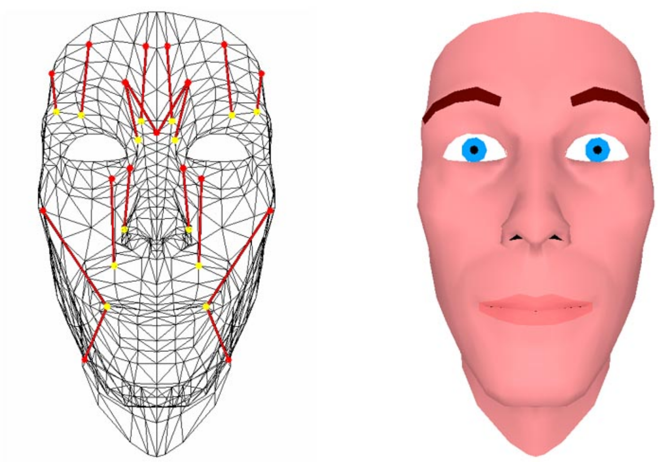

Figure 5: Waters muscle model. The left image show the model mesh and the muscle vectors. The red dots indicate points of origin and yellow dots points of insertion. The fully shaded model is shown on the right.

Each muscle vector has a point of origin, where the muscle would be knitted to the bone, and a point of insertion, where the muscle would be attached to the facial skin. A muscle "contracts" in the direction from the point of insertion to the point of origin.

A zone of influence is defined around each muscle, specified by an angle, α, and radial distances from the origin, fstart and fend (see Figure 6). Each vertex of the mesh within this zone of influence undergoes a displacement towards the muscle origin. The magnitude of the displacement is governed by the muscle contraction factor; the angle that the vertex lies at, relative to the muscle vector; and the distance from the vertex to the muscle origin.

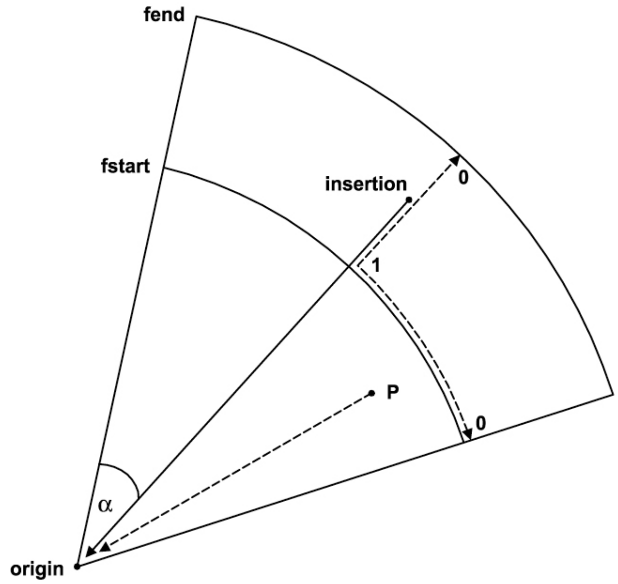

Figure 6: Muscle zone of influence for the Waters muscle model.

There are both angular and radial fall-off factors that affect the calculation of the displacements. The fall-off was designed to mimic the elastic behaviour of the skin. The vertex displacement is multiplied by a factor that scales from 1, for vertices lying along the muscle vector, to 0, for vertices that are at an angle greater than or equal to α.

In addition, vertices lying between fstart and fend use a second scaling factor falling from 1 to 0. Those vertices beyond fend are not displaced at all.

The advantages of this technique are that it simulates muscle contraction and skin movement using a computationally inexpensive routine and it provides good visual results. A more complex method, such as [20], produces better visual results but is much slower.

6.2.2 Our System

We have use Waters' model and modified it slightly for our needs. Our modifications are described in this section.

The original model allows simple jaw rotation and opening of the mouth, but this is not realistic and looks artificial. We have modified the opening of the mouth so that the lips are more rounded and behave more naturally.

In the original model the muscle vectors are defined in terms of absolute positions. In order to facilitate relative movement, we have embedded the muscles into the facial mesh and associated the points of origin and insertion with specific vertices. This allows relative muscle contractions. For example, we can open the model's mouth and then contract the mouth corners into the shape of a smile or frown.

In order to produce an animation we need to apply the tracked expressions to our 3-D model. The normalised feature displacements from the tracking system are specified in 2-D. In order to apply these displacements, we first need to interpret them in terms of the model's conformation. Certain vertices are chosen manually and these specify the conformation information. Once the parameters are in units relative to the model, we can map the displacements to muscle contraction factors.

This mapping is done by first projecting the muscle vectors into 2-D and then finding the component muscle contractions (for each relevant muscle) that will reproduce the desired displacements.

The sequence of steps to produce an expression in the 3-D model is as follows:

1. Interpret the expression parameters in terms of the model's conformation.
2. Use the bottom lip displacement to open the model mouth and rotate the jaw.
3. Project the mouth's muscle vectors to 2-D and determine the relative contractions needed to reproduce the required displacements.
4. Contract the mouth's muscle vectors
5. Repeat steps 3 and 4 for the eyebrow displacements and muscle vectors.

After the expression has been successfully mapped to the 3-D model, the pose information extracted by the tracking system is used to change the model's orientation. This procedure is repeated for each frame of the tracking and animation. Frames may be interpolated for higher frame rates and smoother motion. Figure 7 illustrates the results of animating the 3-D model with tracked expression data.

Presently, the face model consists of a face-mask, eyes, lips and eyebrows. Future enhancements will see the addition of the remainder of the head, neck and hair. Eyelids, teeth and a tongue must also be added. At a later stage we will map textures to the 3-D model for a more realistic avatar.

7 PROTOTYPE SYSTEM

Using the tracking system described earlier and the two 2-D animations, we have built a real-time prototype system. The system tracks the features of an actor, transmits the animation parameters across a network and animates a face on a remote client. The system consists of the following components:

1. The tracking system: the expressions of an actor are tracked and converted into normalised parameters.
2. The communication system: the normalised parameters are quantised and transmitted to remote clients using Windows sockets.
3. The animation system: a remote client receives the animation parameters, interprets them in terms of the model's conformation and produces an animation using either the cartoon face or the morphing algorithm.

The prototype system was demonstrated at a university open day. The following machines were used for the demonstration:

- *Tracking system*: A Dual PII 350 MHz with 256MB RAM, Voodoo 2 and FireGL 1000 Pro graphics cards and Windows 2000. A Creative Web-Cam Go camera provided the video input.
- *Animation system*: A 500 MHz AMD Athlon with 396MB RAM, GeForce 256 graphics card and Windows 2000.

The machines were connected via a T1 LAN and were on separate sub-nets. The tracking and animation systems ran consistently at 13Hz. Occasionally the system mis-tracked due to rapid head movement or disturbance to room lighting. The system has functionality to recover from these situations, however. In extreme cases a user re-initialisation was required.

Most visitors to the demonstration found the animation convincing and entertaining. The cartoon face was surprisingly effective at conveying emotions while the morphed image texture was more visually appealing but less effective. This may have been due to the chosen facial texture, which had no eyebrows and few distinctive facial features.

Noise in the tracking of the markers led to an irritating shaking of facial features in the animation. Subsequent work has reduced this noise but more work is required in this area.

The actress' voice was transmitted and played using a separate audio tool, but visitors found the lack of synchronisation with lip movements disturbing. We plan to incorporate a synchronised audio module into the system in our future work.

Figure 7 (full page after references) shows a few visual results from a live demonstration session. The tracked data was recorded and used later animate the 3-D model seen in the figure.

8 FUTURE TESTS AND EXPERIMENTS

We would like to show that our tracking and animation system faithfully conveys the major facial expressions. To this end, we envisage conducting experiments comparing subjects' recognition of expressions from video and our animations. Schiano et al [28] conducted similar experiments on a simple robot face and showed good recognition rates. We hope to conduct similar tests.

Our major goal is to combine the 3-D model with a full body avatar and integrate the system with a collaborative virtual environment, such as DIVE [5, 6]. We will then conduct a series of experiments to test the effects of real-time facial animation on immersion and virtual "presence".

9 CONCLUSION AND FUTURE WORK

We have presented a system for real-time facial expression tracking and animation. Our system uses relatively low-cost equipment and a set of markers to track facial features. We have used the tracked expressions to animate three different types of faces. We have used a 2-D cartoon face, a morphed 2-D image texture and a more complex 3-D model based on the work of Waters [32].

Our system has been tested with live video input and demonstrated at a university open day. We have shown that low-cost real-time facial animation is feasible and produces satisfactory results. However, future work and user testing is required to fully explore the usefulness of our system.

We list possible future work below:

- We hope to test the reliability of the tracking and animation system by testing if users identify the same expressions from video clips and the corresponding animations.
- Our 3-D model will be improved and then integrated into a collaborative virtual environment. We also need to integrate a synchronised audio module into the animation system. We will then test the effect of real-time facial animation on immersion, presence and the user's emotional investment.
- We hope to reduce the dependence of the tracking system on markers. Ideally we would like to track facial features directly, without any intrusive devices.
- Finally, we envisage alternative applications for our facial feature tracking system. It could be used to adjust computer behaviour to suit the mood of the user, in the spirit of perceptual intelligence [26]. Also, a simple user interface could be built that uses the orientation of the user's head and simple gestures for interaction.

10 ACKNOWLEDGMENTS

We thank Alex Kahle and James Turton for the idea of using Beier-Neely morphing for 2-D face animation. We also thank Wendy Preston for being a patient and well-tempered actress during numerous test sessions.

References

[1] Committee draft of iso/iec 14496-2, information technology - coding of audio-visual objects: Video. Annex C contains Face object decoding tables and definitions, 1996.

[2] Thaddeus Beier and Shawn Neely. Feature-based image metamorphosis. In Edwin E. Catmull, editor, *Computer Graphics (SIGGRAPH*

'92 Proceedings), volume 26, pages 35–42, July 1992.

[3] Biovirtual. http://www.biovirtual.com.

[4] Ian Buck, Adam Finkelstein, Charles Jacobs, Allison Klein, David H. Salesin, Joshua Seims, Richard Szeliski, and Kentaro Toyama. Performance-Driven Hand-Drawn Animation. In *Non-Photorealistic Animation and Rendering Symposium*, June 2000.

[5] Carlsson and Hagsand. DIVE - A Multi User Virtual Reality System. In *IEEE Virtual Reality Annual International Symposium*, pages 394–400, September 18-22 1993.

[6] C. Carlsson and O. Hagsand. DIVE - A Platform for Multi-User Virtual Environments. *Computers and Graphics*, 17(6), 1993.

[7] Juan Casanueva. Presence and Co-Presence in Collaborative Virtual Environments. Master's thesis, University of Cape Town, Cape Town, South Africa, April 2001.

[8] Eric Cosatto and Hans Peter Graf. Photo-Realistic Talking Heads from Image Samples. In *IEEE Transactions on Multimedia*, volume 2, pages 152–163, September 2000.

[9] Charles Darwin. *The Expressions of the Emotions in Man and Animals*. Harper Collins, London, 1999.

[10] Digimask. http://www.digimask.com.

[11] P. Ekman. *Handbook of Social Psychophysiology*, pages 143–146. John Wiley, Chichester, 1989.

[12] Paul Ekman and Wallace V. Friesen. *Manual for the Facial Action Coding System*. Consulting Psychologists Press, Inc., Palo Alto, California, 1978.

[13] Paul Ekman et al. Universal and Cultural Differences in the Judgement of Facial Expressions of Emotion. *Journal of Personality and Social Psychology*, 53:712–717, 1987.

[14] Eyematic. http://www.eyematic.com.

[15] FaceWorks. http://www.interface.digital.com.

[16] International Organisation for Standardisation. Coding of moving pictures and audio iso/iec jtc1/sc29/wg11 n2459. Available at: http://www.cselt.stet.it /mpeg/standards/mpeg-4/mpeg-4.htm, October 1998.

[17] Taro Goto, Marc Escher, Christian Zanardi, and Nadia Magnenat-Thalmann. MPEG-4 based animation with face feature tracking. In *Proceedings of Eurographics'99*, 1999.

[18] Brian Guenter, Cindy Grimm, Daniel Wood, Henrique Malvar, and Fredrick Pighin. Making Faces. In *SIGGRAPH 98*, Computer Graphics Annual Conference series, pages 55–66. Addison Wesley, July 1998.

[19] T. Hagen, P. Noot, and H. Ruttkay. Chartoon: a system to animate 2d cartoon faces. In *Proceedings of Eurographics'99*.

[20] Yuencheng Lee, Demetri Terzopoulos, and Keith Waters. Realistic Modeling for Facial Animation. In *SIGGRAPH 95*, Computer Graphics Annual Conference series, pages 55–62. Addison Wesley, August 1995.

[21] D. Liebowitz and A. Zisserman. Metric rectification for perspective images of planes. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 482–488, June 1998.

[22] N. Mangenat-Thalmann, N.E. Primeau, and D. Thalmann. Abstract muscle actions procedures for human face animation. *Visual Computer*, 3(5):290–297, 1988.

[23] A. Mehrabian. Communication without words. *Psychology Today*, 1968.

[24] Maja Pantic and Leon J.M. Rothkrantz. Automatic Analysis of Facial Expressions: The State of the Art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1424–45, 2000.

[25] F.I. Parke. *A Parametric Facial Model for Human Faces*. PhD thesis, University of Utah, Salt Lake City, UT, December 1974. UTEC-CSc-72-120.

[26] Alex Pentland. Perceptual intelligence. *Communications of the ACM*, 43(3):35–44, March 2000.

[27] S.M. Platt. A system for computer simulation of the human face. Master's thesis, The Moore School, University of Pennsylvania, Philadelphia, 1980.

[28] D.J. Schiano, S.M. Ehrlich, K. Rahardja, and K. Sheridan. Face to InterFace: Facial affect in (Hu)Man and Machine. In *CHI 2000 Conference Proceedings*, pages 193–200, 2000.

[29] J. Semple and G. Kneebone. *Algebraic Projective Geometry*. Oxford University Press, 1979.

[30] M. Slater, J. Howell, A. Steed, D-P. Pertaub, M. Garau, and S. Springel. Acting in Virtual Reality. In *ACM Collaborative Virtual Environments*, pages 103–110, 2000.

[31] SmartKiosks. http://www.crl.research.digital.com /projects/kiosk/default.htm.

[32] Keith Waters. A Muscle Model for Animating Three-Dimensional Facial Expression. In Maureen C. Stone, editor, *Computer Graphics (SIGGRAPH '87 Proceedings)*, volume 21 of *Computer Graphics Annual Conference series*, pages 17–24. Addison Wesley, July 1987.

[33] Lance Williams. Performance-Driven Facial Animation. In Forest Baskett, editor, *Computer Graphics (SIGGRAPH '90 Proceedings)*, volume 24 of *Computer Graphics Annual Conference series*, pages 235–242. Addison Wesley, August 1990.

[34] George Wolberg. *Digital Image Warping*. IEEE Computer Society Press, 1990.

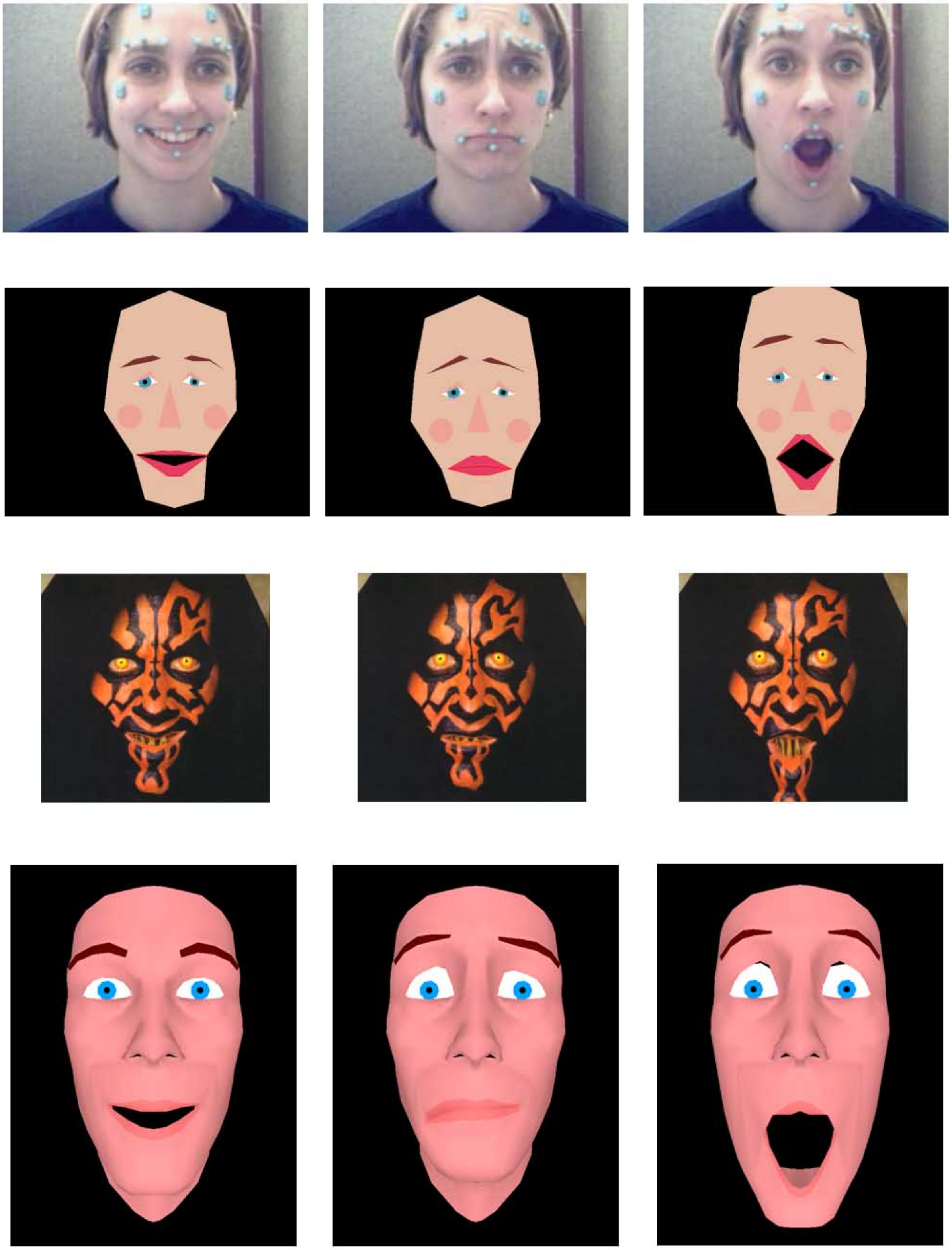

Figure 7: Table of expression correlations for an actress, 2-D cartoon model, 2-D morphed texture and the 3-D muscle model. The 3-D model was animated using data recorded from a demonstration session.