A MULTIMEDIA TESTBED FOR FACIAL ANIMATION CONTROL

PREM KALRA[1]
ENRICO GOBBETTI[2]
NADIA MAGNENAT THALMANN[1]
DANIEL THALMANN[2]

[1] MIRALab, CUI, University of Geneva,
24, rue du General Dufour, 1211 Geneva,
Switzerland
[2] Computer Graphics Lab
Swiss Federal Institute of Technology
1015 Lausanne, Switzerland

ABSTRACT

This paper presents an open testbed for controlling facial animation. The adopted controlling means can act at different levels of abstraction (specification). These means of control can be associated with different interactive devices and media thereby allowing a greater flexibility and freedom to the animator. Possibility of integration and mixing of control means provides a general platform where a user can experiment with his choice of control method. Experiments with input accessories like the keyboard of a music synthesizer and gestures from the DataGlove are illustrated.

Keywords: Facial Expression, Animation Control.

1. Introduction

The human face is an extremely interesting but intricate object. The complexity is not due just to complex anatomical structure but also to muscle actions, dynamics, and psychological and behavioral response. Thus, design of a facial animation system involves several components. In addition these components may include information from different media and sources.

Animating the face manually for every action is a very tedious task and may not even yield the desired results. Faces have their own language where facial actions may express emotions, illustrate speech, comment on feelings or show attitude. Knowledge of such a language and its interaction with the face is essential to improve facial animation system. In addition, understanding of gesture motions of the head and eyes which accompany speech and amplify the communication process is very important.

Most existing and earlier systems[1,8,12,14,18,20,22,26] do not integrate all the aspects in one package. Further, there have not been many experiments which explore various means of specifying facial animation. Where and how the animation characteristics (temporal and spatial) should be specified has been given little attention in the literature. Multi-level

parameterization[19] seems to be a solution for better control. However, simple addition of higher levels of control may not always be an appropriate answer, as the animator may want to obtain some specific features or to perform some low-level 'fine-tuning'. Furthermore, providing manipulation at every level may burden the task of the animator and seemingly obscure the basic goals of the system.

There are presently three main types of facial animation systems in terms of driving mechanism or animation control. One type of systems uses a script or command language for specifying the animation[9,11,15,21]. These systems are simple but non-interactive and thus are not very appropriate for real time animation. In addition, fine-tuning an animation is difficult when merely editing the script, as there exists a non-trivial relation between textual description and animation results. Another type of system is performance-driven where motion parameters are captured from live performance[3,24,27]. These types of motion copying systems are non-flexible and the external control on animation is very limited when used in isolation. Though it provides high accuracy for timings, it is extremely difficult to edit. Systems driven by speech[8,13] are focused in lip-synchronization and speech decomposition into phonemes. These are adequate when animation involves only speech. What would in fact be more desirable, is a system which can encapsulate different kinds of animation specifications and control mechanisms. Such a system would meet the needs of the animator for almost every situation by giving access to the different means of control. The tracking of a live video sequence may provide the basic sequence of a synthetic animation, textual data may produce speech with audio feedback, a hand gesture may govern the gesture motion of the head and eyes, and so on. Here, our attempt is to present how the information from different sources can be related and controlled to give a sequence of animation. As there does not exist what one can refer to as the 'best' framework for motion control for facial animation, this suggests having an open system where one can try several possibilities and chose the one which is subjectively the 'best.'

In order to gain flexibility and modularity in the execution of the system we need a high degree of interaction. We present some of the advanced input accessories which provide natural interaction and thus intuitive control. 3D interaction is already quite popular for many applications, and here we integrate some of the novel paradigms to experiment in the context of controlling facial animation. Possibilities for control with different interactive situations are examined; e.g. gesture dialogue using a DataGlove and musical streams from a MIDI (Musical Instrument Digital Interface)-keyboard. We believe that it is more important to provide a wide range of interaction components than to enforce a particular style of interface. One of the interactive systems for facial expressions presented by deGraf[3] contains the philosophy of using various puppet interfaces to drive facial animation, however, it seems to have hard wiring of devices for manipulations, which restricts flexibility and interchangeability of different device components.

The paper is organized as follows. First we provide the overall structure of the system in Section 2. Section 3 briefly gives the underlying face model. Animation control is presented in the following section. The input accessories developed for facial animation are given in Section 5. A discussion about the system's present status and future enhancement is presented in the subsequent section.

2. System Architecture

Facial movements, like other body movements rely on perception-driven behaviors. Cognitively, these can be understood as externalization or manifestation of verbal or non-verbal communication agents on a face. These agents activate certain channels of a face

associatively which in turn triggers the relevant muscles and which eventually deforms the face. In a computational model, such a behavior can be interpreted as translating behavioral or cerebral activity into a set of functional units which embody the necessary activity-information. The resulting actions are then combined in a sequence of discrete actions which when applied cause the necessary movements on the face. In our system we model such a behavior by separating facial animation into three major components, namely face model, animation controls and composer.

The face model primarily describes the geometric structure of the face, deformation controller and muscle actions. The model receives streams of actions to perform. These actions are decomposed into the required muscle actions and a new instance of the face is derived for each frame.

The animation controls specify animation characteristics[16]. A facial animation system needs to incorporate adequate knowledge about its static and dynamic environments to enable animators to control its execution with (maybe) predefined, yet flexible set of commands. The system's structure therefore should embed such a know-how in a natural way. In order to satisfy this need, our system employs hierarchical structure and modular design. Fig 1. shows the system's basic structure. Commands to the top level of the system need not be detailed descriptions of movement, instead these are like task description -- for example: SAY "I won't go" while LOOKING left-right. The levels underneath are like functional synergies[28] where the task description is processed after having been decomposed into relevant low level motion parameters. The task description contains higher level abstraction entities such as emotion, head motion and speech.
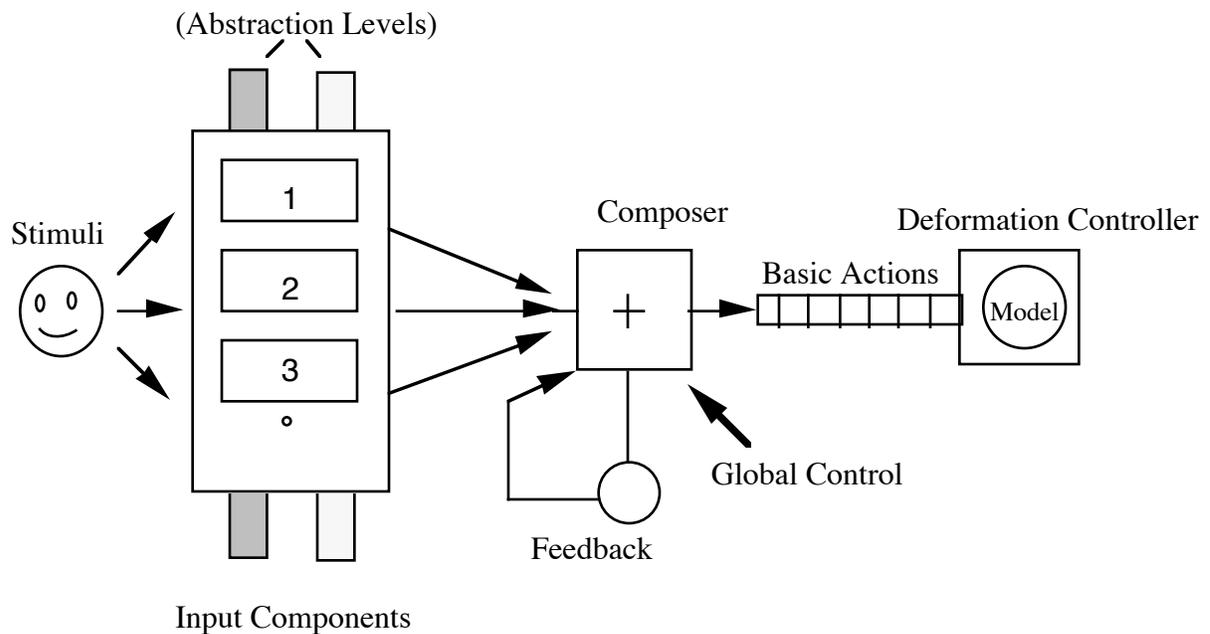

Fig. 1 System's Basic Structure.

The composer acts like a multiplexer and performs integration of the animation controls coming from different sources in (almost) real time. It transmits streams of performable actions to the face model for each time interval.

The animation controls may be driven by different types of input accessories. This provides ease of control specification and handling. The system also allows parallel execution of various operations (tasks); for example, the deformation controller and composer may run in parallel. Similarly, input accessories run independently and produce controlling attributes which are fed to the composer. The system is modular which makes it easy to change certain aspects of the system or to use predefined methods used in other applications. It thereby allows the interchangeability of the input accessories and permits evaluation of their applicability.

The major structural components are described in the following three sections. The first component deals with the facial model, basically addressing the static environment of the system. One can conceive this as a kind of machine-level end or kernel for the system.

3. Face Model

In our facial model the skin surface of a human face, an irregular structure, is considered as a polygonal mesh. Muscular activity is simulated using rational free form deformations[10]. To simulate the effects of muscle actions on the skin of a human face, we define regions on the facial mesh corresponding to the anatomical description of the regions of face where a muscle action is desired. A control lattice can then be defined on the region of interest. The deformations obtained by actuating muscles to stretch, squash, expand and compress the inside volumes of the facial geometry are simulated by displacing the control points of the control lattice. The region inside the control lattice deforms like a flexible volume, according to the displacement and the weight at each control point. Isolating muscle actions from the 3D facial topology avoids the hard-wiring of performable actions. Development of a muscle process that is non-specific to a facial-topology and can be controlled by a small number of parameters provides a more general approach for the modeling and animation of the primary facial expressions.

We use what we call a Minimal Perceptible Action[9] (MPA) as a basic facial motion parameter. Each MPA has a corresponding set of visible movements of the eyebrows, jaw, or mouth and others occurring as a result of muscles contracting and pulling. MPAs also include non-facial muscle actions such as nodding and head turning and movement of the eyes. An MPA can be considered as an atomic action unit similar to the AU (Action Unit) of FACS (Facial Action Coding System)[5], execution of which results in a visible and perceptible variation of a face. We can aggregate a set of MPAs and define expressions and phonemes in our system as facial snapshots, i.e., a particular position of the face at a given time. For phonemes, only the lips are considered during the emission of sound.

The face deformation controller for the face model is a process which waits for each frame described as a sequence of MPAs with their respective intensities and executes the frame by rendering the face at each time interval. The mutual independence and orthogonality of deformation parameters do not force any temporal ordering or sequencing in the execution of the set of MPAs being fed.

4. Animation Controls

Creation of animation is based on the simple principle of motion development which basically needs to provide a temporal sequence of spatial changes. For example, an

animation can be produced by simply providing a stream of MPAs to the deformation controller. This however, does not address the mechanism of motion control. Animation control, in general refers to the specification and manipulation of spatial and temporal characteristics for the various entities involved in the process of animation at different levels.

Animation at the lowest level is specified as a sequence of MPAs with their intensities and time of occurrence. This is an explicit one-level control scheme. The discrete action units defined in terms of MPAs can be used as fundamental building blocks or reference units for the development of a parametric facial process. The control of facial muscles based on MPAs provides a better tool for the creation of grouped functional synergies which avoids the direct control of individual muscles. However, the type of control offered by MPAs is tedious and not adequate enough from an animator's point of view.

Specification of animation in terms of expressions and phonemes makes unnecessary manipulation of MPAs at the lower level. The animator has to define expressions with their respective intensities and duration. The effective intensity of each MPA contained in an expression is influenced by the global intensity of the expression. Animation control using expressions and phonemes is still not natural enough to specify. The animator would rather like to specify and control the animation at the task level where he can describe animation directly in terms of its inherent sources. Thus, in our system we allow the animator to specify animation directly in terms of emotions, head movements and sentences for speech[9]. Such control saves an animator from learning the detailed knowledge of lower-level entities in the system.

The possibility of describing the dimensions of facial animation separately enables us to decompose relevantly a facial animation along a number of independent performable actions. The final decomposition is done at the level of MPAs. The final animation is obtained by summing together the individual sets of actions. In addition, considering different dimensions for facial movements offers the possibility of investigating the meaning and significance of each entity for a given array of animation. This can be easily accomplished by turning off one of the component channels and visualizing the resulting animation. For example, a particular emotion or gesture motion of head alone could be turned on/off to examine its effect on the overall animation.

5. Composer

When an event elicits more than one feeling, emotional blend may occur. For example a person can be surprised and frightened at the same time. Such phenomenon happens when two or more emotions appear simultaneously or when emotions succeed each other rapidly. The measure (strong or feeble) of emotion controls not only the amount of movement but also the appearance of some movements. Therefore blending could also mean blurring the movements. There could also be blending of an emotion, head movements and concurrent speech, all sharing the same MPAs. This can be interpreted when the same type of signal is coming from various channels and one wants to know the resulting amplitude of the signal at that instance of time.

There does not exist a physiological model for blending. However, for emotion blending Ekman and Friesen[4] suggest superimposition of characteristic masks for certain regions of the face, part of the face shows one emotion and part shows another. For example, a blend of surprise and fear is achieved by combining the activities of eyebrows and eyes from the emotion "surprise" and the activities of lower face from "fear". This can be considered as mask-blending. In our model this is realized by providing the facility to

activate or deactivate the different regions like eyes, eyebrows, mouth, jaw etc. Fig 2 shows mask-blending for the emotion "surprise" and "fear".

For blending at MPA level which merges emotion, head motion and speech we employ an adhoc method. Blending here though similar to a union operation, is not simply the summation of individual MPAs. The direct sum may yield values exceeding the extreme limits. For composition of MPAs, we perform summation of the respective MPAs intensities and normalize the sum using a trigonometric sine function. Fig 3 shows the blending function, $S(i)$ refers to the sum of intensities and $R(i)$ to the resulting intensity. Such a composition in a way corresponds to the behavior of neurons where input stimulies are composed with a weighted sum followed by a non-linear transformation function.
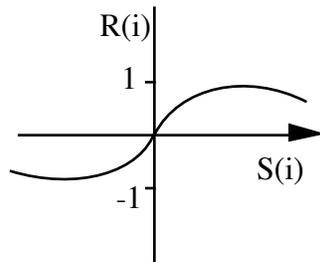
Fig. 3: Blending function.

The composing or blending of actions is performed by Composer. Composer can also be considered as a control manager. Its function is to accept task descriptions in terms of a set of MPAs and build an MPA queue to feed to the face model. The Composer can integrate the controls coming from different sources.

Global control is introduced to affect overall changes at the time of the composition. For example, resulting MPAs may be accentuated or delineated by multiplying their intensities by an appropriate factor. In fact, each input component can be attributed with an attached weight factor which can be changed to suitably affect the output stream of MPAs from that input component (see Fig 4). This yields better overall control. The following section demonstrates the input components used in our system with respect to their potential as control means.
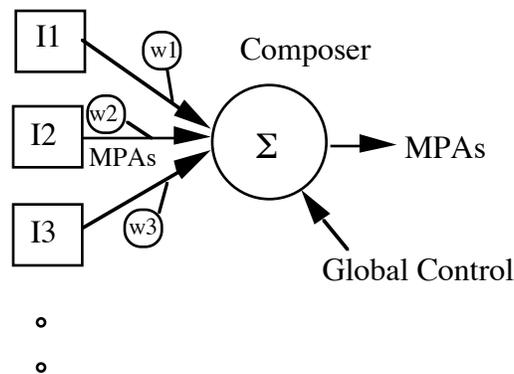
Fig. 4: Composition and global control.

6. Input Accessories

As the system considers the specification of different input components separately from the animation, we can try various ways of controlling the animation. Also, various input components can be used at the same time. Such an approach provides a platform where we can experiment with new methods of control. This analysis may allow us to identify the kind of access we may require for defining and controlling varying levels of abstractions for computational animation models in virtual environments. These accessing components may demand different kinds of interaction which may establish the need for experimentation with several types of devices. As no single mode of control can give completeness, such a test-bed environment can evaluate what device can be used for which means of control. Possibility of composing and mixing different types of controls enhances the reconfigurability of the system. This also allows cooperative group tasks for animation, where more than one person can control the animation in real time. We present here three types of input accessories we have tried. All have some advantages and disadvantages. As there could be simultaneous use of different types at the same type we can overcome some of the disadvantages.

6.1 Script

Script is a standard method of specifying animation consisting of different types of entities. It is like a special language using a few key words for specific operations. Most automated facial animation systems employ this approach.

In our system a language HLSS (High Level Script Scheduler)[9] is used to specify the synchronization in terms of an action and its duration. From action dependence, the starting and the ending times of an action can be deduced. The general format of specifying an action is as follows: while <duration> do <action>. The duration of an action can be a default duration, a relative percentage of the default duration, an absolute duration in seconds, or deduced from other actions preceding or succeeding the present action. The starting time of each action can be specified in different ways, for example, sequentially or parallel using the normal concepts of "fork" and "end" employed in scheduling problems.

One of the major advantages of such an input accessory is that it is in text form. Users can very conveniently change it by editing a text file. On the other hand, being non-interactive it is not possible to change certain parameters while the script is running. Therefore, it is not very suitable for real time animation. It is more useful for background animation.

6.2 MIDI-Keyboard

MIDI-keyboards can be another type of source for controlling the animation. The keyboard has a number of keys enabling us to associate several parameters with the keys. Activation of each key gives two kinds of information: initial velocity with which the key is hit and the pressure variation. System G[6] a real-time, video animation system uses Korg M1 keyboard to move different parts of the mask of a face. However, the system has special purpose processing hardware to perform the animation.

In our system, this device can be used as direct manipulator for MPAs, each key may be assigned to an MPA and the initial velocity of the key may be attached to its intensity. This type of control is at a rather low level. Higher level control can also be obtained by assigning the keys to expressions and phonemes. Fig 5 shows a sequence of facial

animation using the key presses of the MIDI-keyboard. Here, some of the expressions and phonemes have been associated with particular keys of the keyboard, the intensity of the expression or phoneme is governed by the velocity of the key hit. The duration of an expression can be determined by the duration for which the key is pressed. That means an expression starts with the intensity corresponding to the velocity of the key hit and continues until the key is released. At present, the pressure variation of the key hit is not being used, however, we intend to include it to modulate the intensity of an expression during its execution. Sound output from MIDI can provide the needful feedback at the execution of an action. At expression level, it may reflect its intensity. It may be interesting to use array of keys to control a single emotion, here, the mapping of keys would be with the included expression instances or channels, for example, an emotion containing eye, head and mouth motion, each may be associated with a set of keys respectively.

The advantage of such a device is that it provides a number of keys which can be assigned individually to the motion parameters. This gives simultaneous control on many parameters. But at the same time it demands hardwiring of the meaning of desirable action with a particular key. Each time a modification or an extension in the control method will require reconfiguration of the meaning of keys. Also, unidimensional arrangement of keys in the keyboard forces a certain type of order in the manner of control. For music the ordering is with respect to frequency. However, it is not evident how to arrange facial expressions in a unidimensional fashion, consequently this constrains the use of the device in its natural form.

6.3 Postures and Gestures Dialogue

The type of control presented in the previous section is directly dependent on the physical structure of the device used: the mapping between a user's actions and animation controls is obtained by associating a meaning to the various key presses. This device dependency is a factor that limits the animator's expressiveness.

The use of devices which simply sense user's motions, and the use of adaptive pattern recognition can overcome these problems. This gives more freedom to the animator and allows the definition of the mapping between sensor measurements and interpretations to be more complex. Hand gesture recognition is a domain where these techniques can be used. The use of hand-gestures can provide non-verbal cues for a natural human-computer interaction.

In our system, we use posture recognition on data obtained from the DataGlove to obtain categorical and parametric information to drive facial animation. In the ensuing sections we present briefly the posture recognition technique, its continuous classification and application to facial animation control.

6.3.1 Recognition Technique

Posture recognition is a multi-class learning problem: this means that the system has to find an approximate definition for an unknown function f mapping a feature vector x to its class $f(x)$, given a set of training examples of the form $\{x[i], f(x[i])\}$, where the $x[i]$ represents vectors containing the various angles of flexion of the fingers. It therefore involves extraction of invariant feature vectors from the raw sensor data, and the definition of the right mapping function.

The recognition technique here is based on multi-layer perceptrons (MLPs)[23], a type of artificial neural network which is potentially able to approximate any real function[2]. A multi-layer perceptron consists of several layers of processing units with multiple inputs

and one single output which are connected by weighted links. During training each output unit is taught to discriminate between its associated class and all the others by presenting a binary target value equal to +1 if the current input is a member of the class and -1 if not. For training we use a scaled conjugate gradient algorithm[7] to minimize the objective function. Once learning is completed, the weights are frozen and classification is done corresponding to the most active output unit for each of the patterns presented to the network.

6.3.2 Continuous Classification

In order to use the classifier for continuous recognition we use activation thresholds and multiple networks trained on the same examples from different initial conditions. When no posture is currently selected, the system waits until the hand fingers are considered "steady." We compare the maximum instantaneous flex speed with a user-defined threshold. When the hand is considered steady, different networks (usually three) are asked to classify the current hand configuration. A network recognizes a posture only if its most active output unit has a value superior to an upper activation threshold while all the others have their value inferior to a lower activation threshold. The results are then compared and a classification is chosen only if all of the network gives the same response. When a posture is selected, the classifier asks the three networks to classify the data using the same techniques, but with less demanding activation thresholds in order to obtain a stable classification.

6.3.3 Facial Animation Control

Once a posture is recognized, parametric information can be extracted from the location of hand and how it moves. This information can then be used to drive the facial animation. The type of posture as the categorical information can be associated with a type of action performed by the face. We experimented with two types of controls: direct control at the expression level and higher level control at the emotion level.

In the first case, putting the hand in a given posture sets the type of expression for face, while the orientation of the hand controls the expression's intensity. We found this way of control to be adequate for facial editing, but inadequate for real-time animation control.

Control at the higher level, for example the emotion level, is much more appropriate. In this case: putting the hand in a posture selects the type of emotion, while the orientation of the hand at the beginning of the posture with respect to the absolute vertical direction controls the overall intensity of the emotion, and the amount of rotation between the beginning and the end of the posture controls the overall duration. Gesture dialogue for facial animation control provides a continuous dynamic interaction. In this approach one can specify his own definition of a particular gesture by training the gesture recognition system and can associate it with one type of emotion or other task-entity. This gives more flexibility and freedom to the user for controlling the animation.

The position of the hand with respect to a fixed reference frame is normally used to control geometric information such as head and eye rotations. A typical use of this kind of control is to have the synthetic human look at the position of the real hand. Fig 6 shows the example where the movement of the hand correspondingly moves (turns) the head. There does not exist rules to establish the correspondence of a given gesture and an expression, however, experimentation has lead to some natural associations. For example, a gesture from closed fist to completely open hand can indicate the total onset of an expression like "surprise", similarly bending fingers interiorly may be attached to "eyes blinks" or to "head

nodding". The advantage of the gesture recognition system is to have the flexibility of defining ones own gesture for a given expression.

One of the main disadvantages using gesture dialogue as a control means for facial animation is the poor precision offered by the DataGlove. This prohibits the use of this device for fine-tuning. However, for global action manipulation, this provides a natural means of nonverbal communication and interaction method. For duration control of emotion, it is better to associate it with the duration of the gesture itself so as to establish a temporal correspondence between animator's action and the action executed.

7. Discussion

In the preceding section we have described the input accessories for facial animation control showing their positive and negative potentials. Adhering to one type of accessory may not give the suitable control for every situation. Therefore, in order to obtain better animation control, it is desirable to integrate the different means of control. The integration may be at functional unit level where different actions are performed using different input accessories. For example, the head motion may be coming from gesticulation of the DataGlove and other emotions and conversational signals may be coming from the MIDI-keyboard or the script. Consequently, we can combine the positive features of different means of controls. A final script in the form of set of MPAs with respective intensity for each frame can be generated, which may be used later for the production of final animation. There may also be a possibility to have multi-pass system where execution of each pass gives feedback for further refinement and the corresponding change is incorporated in the subsequent pass. For example, there may be a script running with a pre-defined animation and at certain point, a modification is required for the running action. A change in this action may be initiated by another input accessory and the output script is now a modified script of animation. This script when running in the next pass would render the desired animation. This provides incremental refinement of animation without going back to redefine the definition of involved entities. Independent execution of input accessories enables multi-user control at the same time, each user can control one type of accessory and manipulate the relevant functions associated with it.

The system described is an experimental software still under development. The animation results obtained have still limitations on speed and the type of control. The control for fine-tuning is restricted. Possibility to reconfigure and redefine the higher level entities in terms of set of lower level actions using multiple-pass concept will give richer vocabulary for the process of control. We have employed blending of actions only in the spatial context, we propose to include temporal blending where the history of an action will affect the score of its current status. We can also use live video performance to generate synthetic facial animation[17]. We intend to integrate this in the system. A live video performance can provide fast the temporal characteristics for an animation template. This template may then be modified, enhanced and complemented as per the requirements using other accessories. Integration of voice and speech recognition will provide another dimension in the control mechanism.

8. Implementation

The System is written in C with the interface built on top of the Fifth Dimension Toolkit[25]. The various input components are independent processes running on UNIX workstations. The communication between the processes is done through sockets in stream

mode (sequenced two-way communication based on byte streams) using the Internet protocol. Figure 7 shows the components of a proposed system where each component is a different 3D device or a media. The input components communicate with the central process through inter process communication (IPC). A command IPC-server starts the server on the machine, the command is executed. The server basically accepts and distributes messages to all its clients. The output of each component is further processed by IPC-filter to get the desired mapping between the IPC outgoing messages of each components to the incoming action streams to the central process. The central process basically composes the IPC messages coming from different sources and produces the relevant stream of actions (MPAs) to be performed by the face model. Sensory feedback can be provided to users by different means, e.g. real time animation serves as the visual feedback, MIDI output may provide audio feedback and a text output may provide the final animation sequence script. IPC protocol allows the flexibility of developing each component individually and then hook up with the main process.
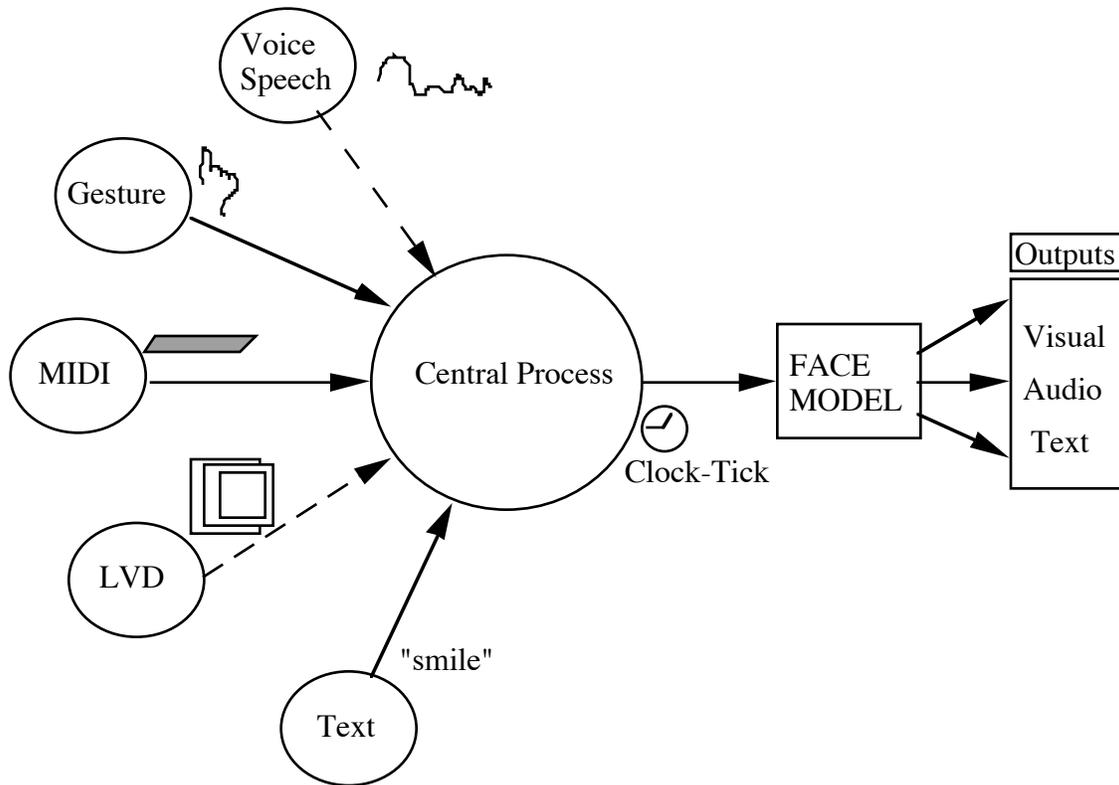

Fig. 7: Proposed system with multimedia components for facial animation control.

9. Conclusion

Face is a multisignal and multimessage system[5]. To interact with such a system we need multiple channels of inputs. We present an open hybrid system for facial animation. It

encapsulates a considerable amount of information regarding facial models, movements, expressions, emotions and speech. The complex description of facial animation can be handled better by assigning multiple input accessories. These input accessories may be a simple script or a multi-input musical keyboard or a gesture dialogue from the DataGlove or some other type of interactive physical or virtual device. Integration of all means of control offers flexibility and freedom to the animator. The scope of such an open system is tremendous. Virtual worlds would be desolate indeed without things like synthetic faces that we can relate to and understand.

Acknowledgments

References

1. Bergeron P, Lachapelle P (1985) *Controlling Facial Expressions and Body Movements in the Computer Generated Animated Short "Tony de Peltrie"*. SIGGRAPH '85 Tutorial Notes, Advanced Computer Animation Course.
2. Cybenko G (1989) *Approximation by Superposition of a Signoidal Function*, Math, Control Signals and Systems (2), pp. 303-314.
3. deGraf B (1989) in *State of the Art in Facial Animation*, SIGGRAPH '89 Course Notes No. 26, pp. 10-20.
4. Ekman P, Friesen WV (1975), *Unmasking the Face: A Guide to Recognizing Emotions from Facial Clues*, Printice-Hall
5. Ekman P, Friesen WV (1978), *Facial Action Coding System*, Investigator's Guide Part 2, Consulting Psychologists Press Inc.
6. Gross D (1991), *Merging Man and Machine*, Computer Graphics World, Vol. 14, No. 5, pp. 47-50.
7. Hestenes M (1980), *Conjugate Direction Methods in Optimization*, Springer Verlag, New York.
8. Hill DR, Pearce A, Wyvill B (1988), *Animating Speech: An Automated Approach Using Speech Synthesized by Rules*, The Visual Computer, Vol. 3, No. 5, pp. 277-289.
9. Kalra P, Mangili A, Magnenat-Thalmann N, Thalmann D (1991) *SMILE : A Multilayered Facial Animation System*, Proc IFIP WG 5.10, Tokyo, Japan (Ed Kunii Tosiyasu L) pp. 189-198.
10. Kalra P, Mangili A, Magnenat-Thalmann N, Thalmann D (1992) *Simulation of Muscle Actions using Rational FreeForm Deformations*, Proc. Eurographics '92, Computer Graphics Forum, Vol. 2, No. 3, pp. 59-69.
11. Kaneko M, Koike A, Hatori Y (1992) *Automatic Synthesis of Moving Facial Images with Expression and Mouth Shape Controlled by Text*, Proc CGI '92, Tokyo (Ed T L Kunii), pp. 57-75.
12. Lewis JP, Parke FI (1987), *Automated Lipsynch and Speech Synthesis for Character Animation*, Proc. CHI '87 and Graphics Interface '87, Toronto, pp. 143-147.
13. Lewis JP (1992), *Automated Lipsynch: Background and Techniques*, The Journal of Visualization and Computer Animation, Vol. 2, No. 4, pp. 118-122, 1992.

14. Magnenat-Thalmann N, Thalmann D (1987), *The Direction of Synthetic Actors in the film Rendez-vous à Montréal*, IEEE Computer Graphics and Applications, Vol. 7, No. 12, pp. 9-19.
15. Magnenat-Thalmann N, Primeau E, Thalmann D (1988), *Abstract Muscle Action Procedures for Human Face Animation*, The Visual Computer, Vol. 3, No. 5, pp. 290-297.
16. Magnenat-Thalmann N, Thalmann D (1991), *Complex Models for Visualizing Synthetic Actors*, IEEE Computer Graphics and Applications. Vol. 11, No. 5, pp. 32-44.
17. Magnenat-Thalmann N, Cazedevals A, Thalmann D (1993), *Modelling Facial Communication between an Animator and a Synthetic Actor in Real Time*, Proc Modeling in Computer Graphics, Genova, Italy (Ed Falcidieno B and Kunii TL) pp. 387-396.
18. Parke FI (1982), *Parametrized Models for Facial Animation*, IEEE Computer Graphics and Applications, Vol. 2, No. 9, pp. 61-68.
19. Parke FI (1991), *Control Parameterization for Facial Animation*, Proc. Computer Animation '91, Geneva, Switzerland, (Eds Magnenat-Thalmann N and Thalmann D), pp. 3-13.
20. Pearce A, Wyvill B, Hill DR (1986), *Speech and Expression: A Computer Solution to Face Animation*, Proc. Graphics Interface '86, Vision Interface '86, pp. 136-140.
21. Pelachaud C, Badler NI, Steadman M (1991), *Linguistic Issues in Facial Animation*, Proc. Computer Animation '91, Geneva, Switzerland, (Eds Magnenat-Thalmann N and Thalmann D), pp. 15-30.
22. Platt S, Badler NI (1981), *Animating Facial Expressions*, Proc SIGGRAPH '81, Computer Graphics, Vol. 15, No. 3, pp. 245-252.
23. Rumelhart D. E., Hinton G. E., Williams R. J. (1986), *Learning Internal Representations by Error Propagation.* In Rumelhart D. E., McClellend J. L. (eds.), Parallel Distributed Processing, Vol. 1: 318-362.
24. Terzopoulos D, Waters K (1991) *Techniques for Realistic Facial Modeling and Animation*, Proc. Computer Animation '91, Geneva, Switzerland, (Eds Magnenat-Thalmann N and Thalmann D), pp. 59-74.
25. Turner R, Gobbetti E, Balaguer F, Mangili A, Thalmann D, Magnenant-Thalmann N (1990), *An Object-Oriented Methodology Using Dynamic Variables for Animation and Scientific Visualization*, Proc CGI 90, Singapore, (Eds Chua TS, Kunii TL), pp. 317-328.
26. Waters K (1987), *A Muscle Model for Animating Three Dimensional Facial Expression*, Proc SIGGRAPH '87, Computer Graphics, Vol. 21, No. 4, pp. 17-24.
27. Williams L (1990), *Performance Driven Facial Animation*, Proc SIGGRAPH '90, Computer Graphics, Vol. 24, No. 3, pp. 235-242.
28. Zeltzer D (1982), *Motor Control Techniques for Figure Animaiton*, IEEE Computer Graphics and Applications, Vol. 2, No. 9, pp. 53-59.