M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 247
Appears: IEEE CVPR 1994 Conference, Seattle, Washington, June 1994 (pp. 76-83)

A Vision System for Observing and Extracting Facial Action Parameters

Irfan A. Essa and Alex Pentland
Perceptual Computing Section, The Media Laboratory
Massachusetts Institute of Technology
Cambridge MA 02139, U.S.A.

Abstract

We describe a computer vision system for observing the "action units" of a face using video sequences as input. The visual observation (sensing) is achieved by using an optimal estimation optical flow method coupled with a geometric and a physical (muscle) model describing the facial structure. This modeling results in a time-varying spatial patterning of facial shape and a parametric representation of the independent muscle action groups, responsible for the observed facial motions. These muscle action patterns may then be used for analysis, interpretation, and synthesis. Thus, by interpreting facial motions within a physics-based optimal estimation framework, a new control model of facial movement is developed. The newly extracted action units (which we name "FACS+") are both physics and geometry-based, and extend the well-known FACS parameters for facial expressions by adding temporal information and non-local spatial patterning of facial motion.

1 Introduction

There is a significant amount research on facial expressions in computer vision and computer graphics (see [4] for review). Perhaps the most fundamental problem in this area is how to categorize active and spontaneous facial expressions to extract information about the underlying emotional states? [3]. Ekman and Friesen [6] have produced the most widely used system for describing visually distinguishable facial movements. This system, called the *Facial Action Coding System* or *FACS*, is based on the enumeration of all "action units" of a face which cause facial movements. As some muscles give rise to more than one action unit, the correspondence between action units and muscle units is approximate.

However, the use of such "frozen" action descriptions is unsatisfactory for a system developed to code *movements*. The lack of temporal and detailed spatial (both local and global) information is a severe limitation of the FACS model (see [4]). The goal of the research presented in this paper is to provide a method for extracting an extended FACS model ("FACS+") using a physics-based model of both skin and muscle, driven by optical flow. We will show that our method is capable of very detailed analysis in both time in space with improved accuracy providing the required information to observe coarticulation of expressions resulting in improved modeling of facial motion.

The plan of this paper is to first describe our mathematical formulation, including geometric and physical modeling, the muscle control model, and our system identification and analysis method. We will then describe our experiments, and illustrate them by use of two example expressions. Finally, we will discuss how our results can be used to improve both the temporal and spatial accuracy of the FACS-like models currently used in computer graphics and machine vision.

1.1 Previous Work

There have been several attempts to track facial expressions over time. Terzopoulos and Waters [17] developed a method to trace linear facial features, estimate corresponding parameters of a three dimensional wireframe face model, and reproduce facial expression. A significant limitation of this system is that it requires that facial features be highlighted with make-up for successful tracking. Although active contour models (*snakes*) are used, the system is still passive; the facial structure is passively shaped by the tracked contour features without any active control based on observations.

Mase [10, 11] has introduced another method to track action units using optical flow. The major limitation of this work is that no physical model is employed; the face motion is formulated statically rather than formulated within a dynamic optimal estimation framework. However, the results of this work convince us of the validity of optical flow computation for observing facial motion.

Haibo Li, Pertti Roivainen and Robert Forchheimer [9] describe an approach in which a control feedback loop between computer graphics and computer vision processes is used for a facial image coding system. Their work is the most similar to ours, but both our goals and implementations differ.

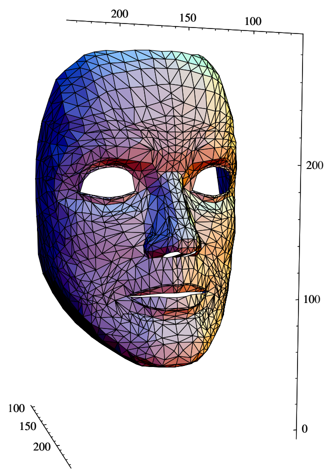


Figure 1: *Geometric Model of a Face (Polygons/Vertices).*

The main limitation of their work is the lack of detail in motion estimation as only large, predefined areas were observed, and only affine motion computed within each area. These limits may be an acceptable loss of quality for image coding applications. However, for our purposes this limitation is severe; it means we cannot observe the "true" pattern of muscle actuation because the method assumes the FACS model as the underlying representation.

2 Vision-based Sensing: Visual Motion

The dynamic evolution of images over time provides enormous amounts of information about a scene. We will therefore use optical flow processing as the basis for perception of facial expressions. We use Simoncelli's [16] method for optical flow computation, which uses a multi-scale, coarse-to-fine, Kalman filtering-based algorithm that provides good motion estimates and error-covariance information. Using this method we compute the estimated mean velocity vector $\hat{\mathbf{v}}_i(t)$, which is the estimated flow from time t to $t+1$. We also store the flow covariances $\Lambda_{\mathbf{V}}$ between different frames for determining confidence measures and for error corrections in observations for the dynamic model (see section 4 and Figure 3 [observation loop (a)]).

3 Facial Modeling

A priori information about facial structure is an important parameter for our framework. Our face model is shown in Figure 1. This is an elaboration of the mesh developed by Platt and Badler [15]. We extend this into a topologically invariant physics-based model by adding

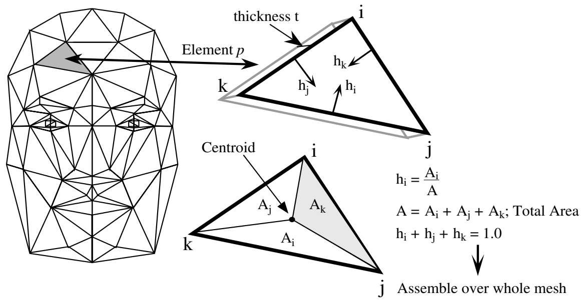


Figure 2: *Using the FACS mesh to determine the continuum mechanics parameters of the skin using FEM.*

anatomically-based muscles to it.

In order to conduct analysis of facial expressions and to define a new suitable set of control parameters ("FACS+") using vision-based observations, we require a model with time dependent *states* and *state evolution* relationships. FACS and the related AU descriptions are purely static and passive, and therefore the association of a FACS descriptor with a dynamic muscle is inherently inconsistent. This problems motivated Waters [18] to develop a muscle model in a dynamic framework. By modeling the elastic nature of facial skin and the anatomical nature of facial muscles he developed a dynamic model of the face, including FACS-like control parameters. By implementing a procedure similar to that of Waters', we also built a dynamic muscle-based model of a face.

3.1 Physically-based Modeling

A physically-based dynamic model of a face requires use of Finite Element methods. These methods give our facial model an *anatomically-based* facial structure by modeling facial tissue/skin, and muscle actuators, with a geometric model to describe force-based deformations and control parameters.

For dynamic modeling we need to integrate the system dynamics with respect to the following equation of rigid and nonrigid dynamics.

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{R}. \tag{1}$$

where $\mathbf{u} = [U, V, W]^T$ is the global deformation vector, which describes the deformation in the facial structure over time. Assuming we use the polygonal mesh shown in Figure 1 as the finite element mesh with n nodes and m elements, then $\mathbf{M}$ is a $(3n \times 3n)$ mass matrix, which accounts for the inertial properties of the face, $\mathbf{K}$ is a $(3n \times 3n)$ stiffness matrix, which accounts for the internal energy due to its elastic properties, $\mathbf{D}$ is a $(3n \times 3n)$ damping matrix, and $\mathbf{R}$ is a $(3n \times 1)$ applied load vector, characterized by the force actuations of the muscles (see [1, 7, 12]).

3.2 Skin and Tissue Modeling

By defining each of the triangles on the polygonal mesh in Figure 1 as an *isoparametric triangular shell element*, (shown in Figure 2), we can calculate the mass, stiffness and damping matrices for each element (using $dV = tdA$), given the material properties of skin. Then by the assemblage process of the direct stiffness method [1, 7] the required matrices for the whole mesh can be determined. As the integration to compute the matrices is done prior to the assemblage of matrices, each element may have different thickness t, although large differences in thickness of neighboring elements are not suitable for convergence [1].

The next step in formulating this dynamic model of the face is the combination of the skin model with a dynamic muscle model. This requires information about the attachment points of the muscles to the face, or in our geometric case the attachment to the vertices of the geometric surface/mesh. The work of Pieper [14] and Waters [18] provides us with the required detailed information about muscles and muscle attachments.

4 Dynamic Modeling and Estimation

4.1 Initialization of FACS/FEM on an image

Extracting data from a vision system and mapping it onto a polygonal mesh requires good initial estimates of structure and location of the polygonal mesh. Consequently initialization of the face template onto an image of a face is an important issue. Currently we manually place the deformable face template to provide an initial guess. After the initial placement, the system can accommodate the deformations and the translations on the basis of its observer mechanics, thus determining the global and local motion in an image.

4.2 Images to FACS, FEM and Muscles

Simoncelli's [16] coarse-to-fine algorithm for optical flow computations provides us with an estimated flow vector, $\hat{\mathbf{v}}_i$. Now using the a mapping function, $\boldsymbol{\mathcal{M}}$, we would like to compute velocities for the vertices of the FACS model $\mathbf{v}_g$:

$$\mathbf{v}_g(x, y, z) = \boldsymbol{\mathcal{M}}(x, y, z)\hat{\mathbf{v}}_i(x, y \,|\, z, y). \tag{2}$$

Then, using the physically-based modeling techniques of section 3.1 and the relevant geometric and physical models we can calculate the forces that caused the motion. Since we are mapping global information from an image (over the whole image) to a geometric model, we have to concern ourselves with translations (vector $\boldsymbol{\mathcal{T}}$), and rotations (matrix $\boldsymbol{\mathcal{R}}$). The Galerkin polynomial interpolation function $\mathbf{H}$ and the strain-displacement function $\boldsymbol{\mathcal{B}}$, used to define the mass, stiffness and damping matrices (in Equation (1)) on the basis of the finite element method are applied to describe the deformable behavior of the model [7, 13, 1].

We would like to use only a frontal view to determine and model expressions, and this is only possible if we are prepared to estimate the velocities and motions in the third axis (going into the image, the z-axis). Using the convex nature of the face, we can determine some of the deformations in the third axis too. We define a function that does spherical mapping, $\boldsymbol{\mathcal{S}}(u, v)$, where are u and v are the spherical coordinates. The spherical function is computed by use of a prototype 3-D model of a face with a spherical parameterization; this canonical face model is then used to wrap the image onto the shape. In this manner, we determine the mapping equation:

$$\mathbf{v}_g(x, y, z) \approx \mathbf{H}\boldsymbol{\mathcal{S}}\boldsymbol{\mathcal{R}}\left(\hat{\mathbf{v}}_i(x, y) + \boldsymbol{\mathcal{T}}\right). \tag{3}$$

For the rest of the paper, whenever, we talk about velocities, we will assume that the above mapping has already been applied, unless otherwise specified.

4.3 Estimation and Control

Estimating motion and then driving a physical system with the inputs from such a noisy source is prone to errors, and can result in divergence or a chaotic physical response. This is why an estimation and control framework needs to be incorporated to obtain stable and well-proportioned results. Similar considerations motivated the control framework used in [9]. Figure 3 shows the whole framework of estimation and control of our active facial expression modeling system. The next few sections discuss these formulations.

4.4 Dynamic system with noise

In a dynamic system with state $\mathbf{X}$, control input vector $\mathbf{U}$, and measurement vector $\mathbf{Y}$, we may write the dynamic system in continuous time state-space form:

$$\dot{\mathbf{X}}(t) = \mathbf{A}\mathbf{X}(t) + \mathbf{B}\mathbf{U}(t) + \mathbf{G}\mathbf{n}_p(t), \tag{4}$$

where the matrix $\mathbf{G}$ defines the coupling of the noise process $\mathbf{n}_p$ with the dynamic system. This is known as a *dynamic state evolution equation* or a *process equation*. In our system the state is positions, velocities and accelerations of nodal points, and control is muscle actuations.

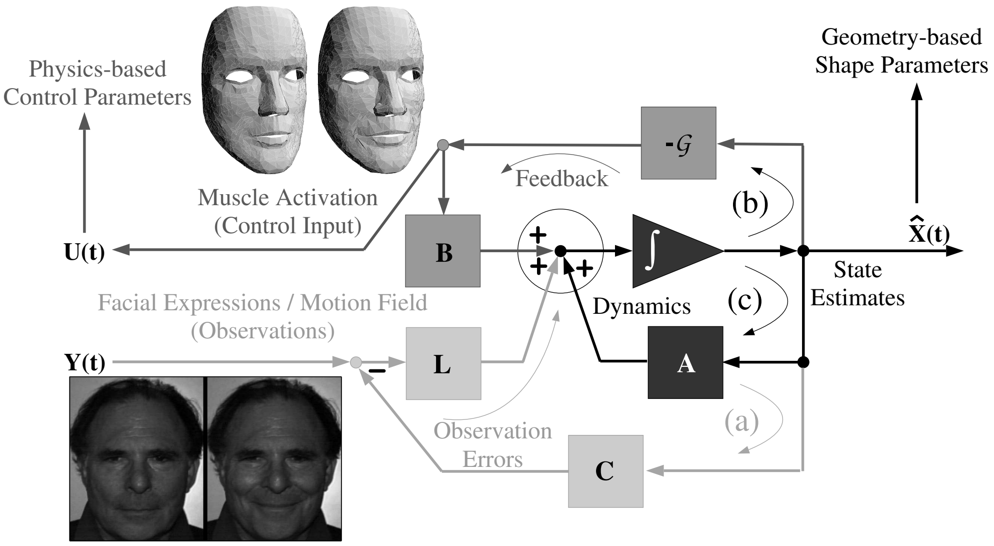


Figure 3: *Block diagram of the proposed control-theoretic approach. Showing the estimation and correction loop (a), the dynamics loop (b), and the feedback loop (c).*

Observations and measurements are modeled by *measurement equation*, where the measurements $\mathbf{Y}$ are determined from states $\mathbf{X}$, and inputs $\mathbf{U}$:

$$\mathbf{Y}(t) = \mathbf{C}\mathbf{X}(t) + \mathbf{D}\mathbf{U}(t) + \mathbf{n}_m(t), \tag{5}$$

where $\mathbf{n}_m$ is the measurement noise. In our model, $\mathbf{D} = 0$, since throughout our observations there is no direct relationship between control and observations. We assume that $\mathbf{n}_p$ and $\mathbf{n}_m$ are white noise processes with covariances $\boldsymbol{\Lambda}_p$ and $\boldsymbol{\Lambda}_m$, respectively. We also assume that both $\mathbf{n}_p$ and $\mathbf{n}_m$ are *uncorrelated zero mean white noise* processes, and that the initial state is also uncorrelated with the measurement and process noises.

4.5 Prediction, Estimation, and Correction

The continuous time Kalman filter (CTKF) allows us to estimate the uncorrupted state vector, and produces an *optimal least-squares estimate* under quite general conditions [2, 8]. The Kalman filter is particularly well-suited to this application because it is a recursive estimation technique, and so does not introduce any delays into the system (keeping the system active). The CTKF for the above system is:

$$\dot{\hat{\mathbf{X}}} = \mathbf{A}\hat{\mathbf{X}} + \mathbf{B}\mathbf{U} + \mathbf{L}\left(\mathbf{Y} - \mathbf{C}\hat{\mathbf{X}}\right), \tag{6}$$

where $\hat{\mathbf{X}}$ is the linear least squares estimate of $\mathbf{X}$ based on $Y(\tau)$ for $\tau < t$. Let $\boldsymbol{\Lambda}_e$ be the error covariance matrix for $\hat{\mathbf{X}}$ then:

$$\mathbf{L} = \boldsymbol{\Lambda}_e \mathbf{C}^T {\boldsymbol{\Lambda}_m}^{-1}, \tag{7}$$

is the Kalman gain matrix. The Kalman gain matrix $\mathbf{L}$ is obtained by solving the following Riccati equation to obtain the optimal error covariance matrix $\boldsymbol{\Lambda}_e$:

$$\frac{d}{dt}\boldsymbol{\Lambda}_e = \mathbf{A}\boldsymbol{\Lambda}_e + \boldsymbol{\Lambda}_e\mathbf{A}^T + \mathbf{G}\boldsymbol{\Lambda}_p\mathbf{G}^T - \boldsymbol{\Lambda}_e\mathbf{C}^T{\boldsymbol{\Lambda}_m}^{-1}\mathbf{C}\boldsymbol{\Lambda}_e. \tag{8}$$

The Kalman filter, Equation (6), mimics the noise free dynamics and corrects its estimate with a term proportional to the difference $(\mathbf{Y} - \mathbf{C}\hat{\mathbf{X}})$, which is the innovations process. This correction is between the observation and our best prediction based on previous data. Figure 3 shows the estimation loop (the bottom loop) which is used to correct the dynamics based on the error predictions.

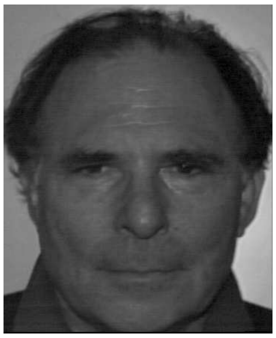
The optical flow computation method has already established a probability distribution $(\boldsymbol{\Lambda}_{\mathbf{V}}(t))$ with respect to the observations. We can simply use this distribution in our dynamic observations relationship of Equation (5). Hence using the mapping criteria as discussed earlier we get:

$$\begin{aligned} \boldsymbol{\Lambda}_m(t) &= \boldsymbol{\mathcal{M}}(x, y, z)\boldsymbol{\Lambda}_{\mathbf{V}}(t), \\ \mathbf{Y}(t) &= \boldsymbol{\mathcal{M}}(x, y, z)\hat{\mathbf{v}}_i(t) \end{aligned}$$

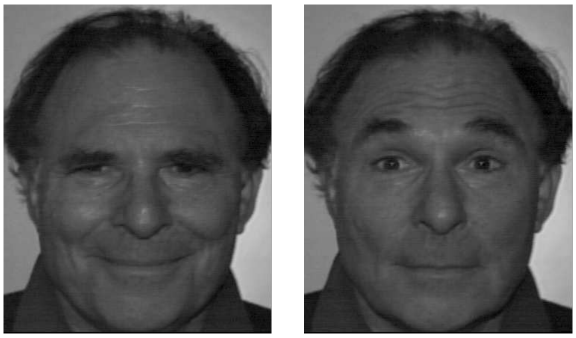
4.6 Control of Dynamic Motion

Now using a control theory approach we will obtain the muscle actuations. These actuations are derived from the observed image velocities. The control input vector $\mathbf{U}$ is therefore provided by the control feedback law:

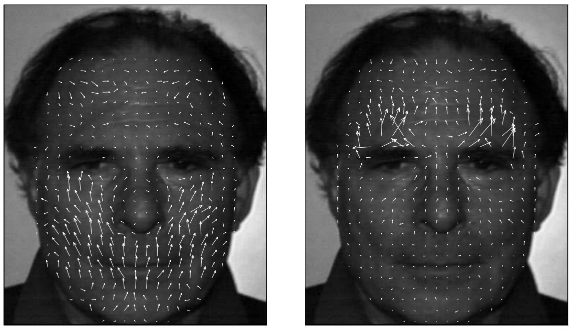
$$\mathbf{U} = -\boldsymbol{\mathcal{G}}\mathbf{X}, \tag{9}$$



(a) Neutral Face



(b) Smile *(c) Raise Brow*



(d) and (e) Motion Fields

Figure 4: *Expressions from video sequences. (a) neutral expression, (b) eighth frame from a smile sequence and (c) the eighth frame from a raising brow sequence, (d) and (e) motion fields for the smile and raising brow expressions*

where $\mathcal{G}$ is the *control feedback gain matrix*. We assume the instance of control under study falls into the category of an *optimal regulator* [8]. Hence, the optimal control law $\mathbf{U}^*$ is given by:

$$\mathbf{U}^* = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}_c\mathbf{X}^* \tag{10}$$

where $\mathbf{X}^*$ is the optimal state trajectory and $\mathbf{P}_c$ is given by solving yet another *matrix Riccati equation* [8]. Here $\mathbf{Q}$ is a real, symmetric, positive semi-definite *state weighting* matrix and $\mathbf{R}$ is a real, symmetric, positive definite *control weighting* matrix. Comparing with Equation (9) we obtain $\mathcal{G} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}_c$ This control loop is also shown in the block diagram in Figure 3 (upper loop (c)).

5 Analysis and Identification

For the synthesis of facial expressions we need appropriate control parameters. In the past this has led to the use of the FACS model. Our method extracts control parameters by observing expressions rather than modeling them *a priori*. These extracted parameters can control much more complex motions than is typical using the FACS and/or muscle models. For instance, experiments described in the next section show that a smile, even though primarily due to actuation of muscles in the lower part of the face, is not complete without some facial deformation in the upper part of the face. This result is corroborated by Ekman [5] who argues that the actuation of upper muscles is a significant part of a "real" smile.

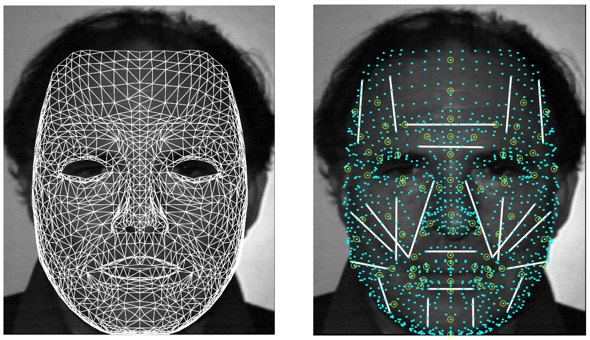
These new control parameters can be described mathematically as follows: Consider a basis set $\boldsymbol{\Phi}_g$, which has p vectors, $\phi_o, \ldots, \phi_p$. Each of the basis vectors, ϕ_i is a deformation profile of a face for a specific action. In a static case, these vectors would typically be FACS-like action units.

The visual observation and estimation process extracts information about the time-evolving deformation profile by extracting a new dynamic basis set $\boldsymbol{\Phi}_g$, using principle component analysis. This new basis set $\boldsymbol{\Phi}_g$, can be used as a "rotation" matrix to "rotate" polygon vertex displacements to a new *generalized set of displacements* $\tilde{\mathbf{u}}$:

$$\mathbf{u} = \boldsymbol{\Phi}_g\tilde{\mathbf{u}} \tag{11}$$

The resulting generalized displacements show distinct characteristics for different expressions. In the range of all expressions, the characteristics of each expression are easily identifiable. Similar results for lip reading and expression recognition using FACS were obtained by Mase and Pentland [10, 11], although within a static estimation framework. Experiments that are described later show distinct deformations and actuations in both space and time for different expressions. This basis set also functions as a set of constraints on the system and is used to determine the control input for different expressions.

Another important transformation is the transformation of the nodal forces $\mathbf{R} = \mathbf{H}\hat{\mathbf{R}}$ to a set of *generalized loads* $\tilde{\mathbf{R}}$. This transformation requires another basis set $\boldsymbol{\Phi}_p$, with each of its vectors defining muscle actuations causing different facial expressions. This *force-based* basis set is obtained by mapping nodal forces and the causal nodal deformations to the parametric representation of muscles



(a) Mesh (b) Muscles

Figure 5: *(a) Face image with a mesh placed accurately over it and (b) Face image with muscles (white lines), region centers (circles) and nodes (dots).*

actuations (rather than just geometric deformations as in the case of deformation basis $\boldsymbol{\Phi}_g$). Let $\mathbf{G}$ be this mapping function, using this mapping we obtain:

$$\boldsymbol{\Phi}_p = \mathbf{G}\boldsymbol{\Phi}_g. \tag{12}$$

Now we can compute the generalized muscle actuations:

$$\mathbf{R} = \boldsymbol{\Phi}_p \tilde{\mathbf{R}}. \tag{13}$$

These basis set, $\boldsymbol{\Phi}_p$ is computed actively during the proposed control loop. The application of such a "near-orthogonal" basis set is less computationally expensive as there are fewer muscle descriptors than geometric descriptors and the muscle descriptors are independent of the topology of the face. Examples of both these types of basis sets are given in the next section.

6 Experiments

The first step in conducting these experiments was to acquire image sequences of subjects making expressions. For this purpose we set up an experimental rig with two calibrated and synchronized cameras to acquire front and side views of a subject. After digitizing the acquired video sequences (Figure 4), optical flow was computed using the coarse-to-fine algorithm [16]. The flow was computed at the full image resolution to ensure that small motions can be detected and accounted for. The results of optical flow computation, for the smiling and raising eyebrows expression, are shown as motion vectors in Figures 4(d) and (e).

In these experiments the geometric model consisted of a 1226 node, 2512 polygon geometric model with 80 facial regions (based on [15], shown in Figure 1). This polygonal model was then used to generate a finite element mesh. The interpolation $\mathbf{H}$ and strain-displacement $\mathcal{B}$ matrices are determined for the given geometry by using the triangular polygons (Figure 2) as two dimensional isoparametric shell elements. Using these interpolation and strain-displacement matrices, the mass, stiffness and damping matrix for each element were computed and then assembled into a matrix for the whole mesh. Material properties of real human skin were used in this computation. These material properties were obtained from Pieper's Facial Surgery Simulator [14]. To this physically-based skin model muscles were attached using the method of Waters [18] and using muscle data from Pieper [14]. This provides an anatomical muscle model of the face, which deforms on actuation of muscles (Figure 5).

To begin the analysis the geometric mesh is initialized by warping it to accurately fit the observed face. Currently this is done by hand. Optical flow is then calculated, as in Figure 4 (d) and (e), and projected onto the mesh. This produces deformation of the skin; the observed spatial deformation for raising eyebrow expression is shown in Figure 6. For comparison, the FACS model for the same motion is also shown. These deformations form the basis set $\boldsymbol{\Phi}_g$ for geometric motion as discussed in last section. Note that since we do have a complete physical model of the skin (deformation surface) available, we can take the spatial analysis of the facial expression into the physical domain. Stress plots of skin deformations were computed and to show detailed descriptions of motion and stress discontinuities resulting in wrinkles and furrows.

Finally, the control-feedback loop (see Figure 3) estimates the muscle control needed to produce the observed temporal and spatial patterning. Figure 7 shows the muscle actuations corresponding to the observed facial motion in the smiling and eyebrow raising cases. These muscle actuations form the basis set $\boldsymbol{\Phi}_p$ for physics-based motion as discussed earlier.

6.1 Analysis

The goal of this work is to develop a new model of facial action ("FACS+") that more accurately models facial motion. The current state-of-the-art for facial description (either FACS itself or muscle-control versions of FACS) has two major weaknesses:

- The action units are purely local spatial patterns. Real facial motion is almost never completely localized; Ekman himself has described some of these action units as an "unnatural" type of facial movement.
- There is no time component of the description, or only a heuristic one. From EMG studies it is known that

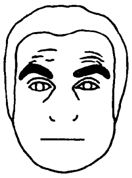
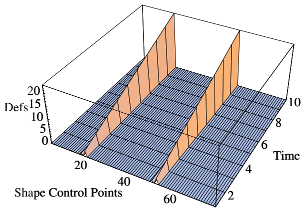
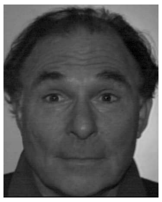
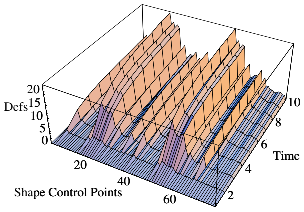
Expression	Magnitude of Control Point Deformation
$AU2$ 	
Raising Eyebrow. 	

Figure 6: *Observed Raising Brows versus static expression: Surface plots showing deformation over time for FACS actions* $AU2$*, and for an actual video sequence of raising eyebrows.*

> most facial actions occur in three distinct phases: *application*, *release* and *relaxation*. In contrast, current systems typically use simple linear ramps to approximate the actuation profile.

Other limitations of FACS include the inability to describe fine eye and lip motions, and the inability to describe the coarticulation effects found most commonly in speech. Although the muscle-based models used in computer graphics have alleviated some of these problems [18], they are still too simple to accurately describe real facial motion.

Consequently, we have focused our efforts on characterizing the functional form of the actuation profile, and on determining an orthogonal basis set of "action units" that better describes the spatial properties of real facial motion. We will again illustrate our results using the smile and eyebrow raising expressions.

Spatial Patterning

The top row of Figure 6 shows $AU2$ ("Raising Eyebrow") from the FACS model and the linear actuation profile of the corresponding geometric control points. This is the type of spatial-temporal patterning commonly used in computer graphics animation.

Below this is shown the observed motion of these control points for the expressions of Raising Eyebrow. As can be seen, the observed pattern of deformation is very different than that assumed in the standard computer graphics implementation of FACS. There is a wide distribution of motion through all the the control points, and the temporal patterning of the deformation is far from linear. By using these observed distributed patterns of motion more realistic image synthesis and computer animations can be produced.

Temporal Patterning

Figure 7 shows plots of facial muscle actuations for the smile and eyebrow raising expressions. In this figure the 24 face muscles were combined into seven local groups for purposes of illustration. As can be seen, even the simplest expressions require multiple muscle actuations.

Of particular interest is the temporal patterning of the muscle actuations. We have fit exponential curves to the activation and release portions of the muscle actuation profile to suggest the type of rise and decay seen in EMG studies of muscles. From this data we suggest that the relaxation phase of muscle actuation is mostly due to passive stretching of the muscles by residual stress in the skin.

Note that Figure 7(b) for the smile expression also shows a second, delayed actuation of muscle group 7 about 3 frames after the peak of muscle group 1. This example illustrates that coarticulation effects can be observed by our system, and that they occur even in quite simple expressions. By using these observed temporal patterns of muscle activation, rather than simple linear ramps, more realistic computer animations and synthetic images can be generated.

7 Conclusions

We have developed a mathematical formulation and implemented a computer vision system capable of detailed analysis of facial expressions within an active and dynamic framework. The purpose of this system to to analyze real facial motion in order to derive an improved model ("FACS+") of the spatial and temporal patterns exhibited by the human face.

This system analyzes facial expressions by observing expressive articulations of a subject's face in video sequences. The visual observation (sensing) is achieved by using an *optimal optical flow* method. This motion is then coupled to a physical model describing the skin and muscle structure, and the muscle control variables estimated.

By observing the control parameters over a wide range of facial motions, we can then extract a minimal parametric representation of facial control. We can also extract a minimal parametric representation of facial patterning, a representation useful for static analysis of facial expression.

Our experiments to date have demonstrated that we can indeed extract FACS-like models that are more accurate than existing models. We are now processing data from

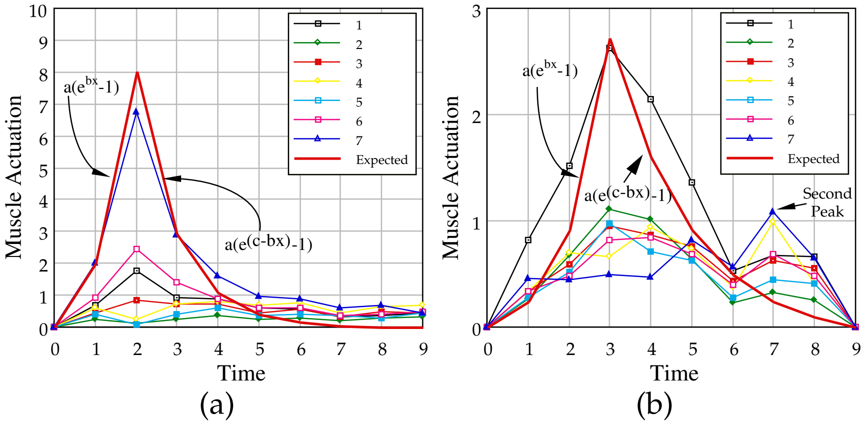


Figure 7: *Muscle actuations over time of the seven main muscle groups modeled for the expressions of (a) raising brow and (b) smile expressions.*

a wider range of facial expression in order to develop a model that is adequate for all facial expressions.

Acknowledgments

We would like to thank Eero Simoncelli and John Wang for all their help with the visual sensing and optical flow code. We would also like to thank Trevor Darrell, Paul Ekman, Steve Pieper, and Keith Waters.

References

[1] Klaus-Jürgen Bathe. *Finite Element Procedures in Engineering Analysis*. Prentice-Hall, 1982.

[2] Robert G. Brown. *Introduction to Random Signal Analysis and Kalman Filtering*. John Wiley & Sons Inc., 1983.

[3] Vicki Bruce. *Recognising Faces*. Lawrence Erlbaum Associates, 1988.

[4] P. Ekman, T. Huang, T. Sejnowski, and J. Hager (Editors). Final Report to NSF of the Planning Workshop on Facial Expression Understanding. Technical report, National Science Foundation, Human Interaction Lab., UCSF, CA 94143, 1993.

[5] Paul Ekman. Facial expression of emotion: An old controversy and new findings. *Philosophical Transactions: Biological Sciences (Series B)*, 335(1273):63–69, 1992.

[6] Paul Ekman and Wallace V. Friesen. *Facial Action Coding System*. Consulting Psychologists Press Inc., 577 College Avenue, Palo Alto, California 94306, 1978.

[7] Irfan A. Essa, Stan Sclaroff, and Alex Pentland. Physically-based modeling for graphics and vision. In Ralph Martin, editor, *Directions in Geometric Computing*. Information Geometers, U.K., 1993.

[8] Bernard Friedland. *Control System Design: An Introduction to State-Space Methods*. McGraw-Hill, 1986.

[9] Haibo Li, Pertti Roivainen, and Robert Forchheimer. 3-d motion estimation in model-based facial image coding. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(6):545–555, June 1993.

[10] Kenji Mase. Recognition of facial expressions for optical flow. *IEICE Transactions, Special Issue on Computer Vision and its Applications*, E 74(10), 1991.

[11] Kenji Mase and Alex Pentland. Lipreading by optical flow. *Systems and Computers*, 22(6):67–76, 1991.

[12] Dimitri Metaxas and Demetri Terzopoulos. Shape and nonrigid motion estimation through physics-based synthesis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(6):581–591, 1993.

[13] Alex Pentland and Stan Sclaroff. Closed form solutions for physically based shape modeling and recovery. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(7):715–729, July 1991.

[14] Steven Pieper, Joseph Rosen, and David Zeltzer. Interactive graphics for plastic surgery: A task level analysis and implementation. *Computer Graphics, Special Issue: ACM Siggraph, 1992 Symposium on Interactive 3D Graphics*, pages 127–134, 1992.

[15] S. M. Platt and N. I. Badler. Animating facial expression. *ACM SIGGRAPH Conference Proceedings*, 15(3):245–252, 1981.

[16] Eero P. Simoncelli. *Distributed Representation and Analysis of Visual Motion*. PhD thesis, Massachusetts Institute of Technology, 1993.

[17] Demetri Terzopoulus and Keith Waters. Analysis and synthesis of facial image sequences using physical and anatomical models. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(6):569–579, June 1993.

[18] Keith Waters and Demetri Terzopoulos. Modeling and animating faces using scanned data. *The Journal of Visualization and Computer Animation*, 2:123–128, 1991.