Chapter 1

Computer Facial Animation: A Survey

Zhigang Deng[1] and Junyong Noh[2]

[1] Computer Graphics and Interactive Media Lab, Department of Computer Science, University of Houston, Houston, TX 77204 zdeng@cs.uh.edu
[2] Graduate School of Culture Technology (GSCT), Korea Advanced Institute of Science and Technology (KAIST), Daejon, Korea junyongnoh@kaist.ac.kr

1 Introduction

Since the pioneering work of Frederic I. Parke [1] in 1972, significant research efforts have been attempted to generate realistic facial modeling and animation. The most ambitious attempts perform the face modeling and rendering in real time. Because of the complexity of human facial anatomy and our inherent sensitivity to facial appearance, there is no real time system that generates subtle facial expressions and emotions realistically on an avatar. Although some recent work produces realistic results with relatively fast performance, the process for generating facial animation entails extensive human intervention or tedious tuning. The ultimate goal for research in facial modeling and animation is a system that 1) creates realistic animation, 2) operates in real time, 3) is automated as much as possible, and 4) adapts easily to individual faces.

Recent interest in facial modeling and animation is spurred by the increasing appearance of virtual characters in film and video, inexpensive desktop processing power, and the potential for a new 3D immersive communication metaphor for human-computer interaction. Much of the facial modeling and animation research is published in specific venues that are relatively unknown to the general graphics community. There are few surveys or detailed historical treatments of the subject [2]. This survey is intended as an accessible reference to the range of reported facial modeling and animation techniques.

Strictly classifying facial modeling and animation techniques is a difficult task, because exact classifications are complicated by the lack of exact boundaries between methods and the fact that recent approaches often integrate several methods to produce better results. In this survey, we roughly classify facial modeling and animation techniques into the following categories: blend shape or shape interpolation (Section 2), parameterizations (Section 3), Facial

Action Coding System based approaches (Section 4), deformation based approaches (Section 5), physics based muscle modeling (Section 6), 3D face modeling (Section 7), performance driven facial animation (Section 8), MPEG-4 facial animation (Section 9), visual speech animation (Section 10), facial animation editing (Section 11), facial animation transferring (Section 12), and facial gesture generation (Section 13). It should be noted that because the facial animation field has grown into a complicated and broad subject, this survey chapter does not cover every aspect of virtual human faces, such as hair modeling and animation, tongue and neck modeling and animation, skin rendering, wrinkle modeling, etc.

2 Blend Shapes or Shape Interpolation

Shape interpolation (blend shapes, morph targets and shape interpolation) is the most intuitive and commonly used technique in facial animation practice. A blendshape model is simply the linear weighted sum of a number of topologically conforming shape primitives (Eq. 1).

$$v_j = \sum w_k b_{kj} \tag{1}$$

In the above Eq. 1, v_j is the j^{th} vertex of the resulting animated model, w_k is blending weight, and b_{kj} is the j^{th} vertex of the k^{th} blendshape. The weighted sum can be applied to the vertices of polygonal models, or to the control vertices of spline models. The weights w_k are manipulated by the animator in the form of sliders (with one slider for each weight) or automatically determined by algorithms [3]. It continues to be used in projects such as *the Stuart Little*, *Star Wars*, and *Lord of the Rings* and was adopted in many commercial animation software packages such as Maya and 3D Studio Max. The simplest case is an interpolation between two key-frames at extreme positions over a time interval (Figure 1.1).

Linear interpolation is often employed for simplicity [4, 5], but a cosine interpolation function [6] or other variations such as spline can provide acceleration and deceleration effects at the beginning and end of an animation. When four key frames are involved, rather than two, bilinear interpolation generates a greater variety of facial expressions than linear interpolation [7]. Bilinear interpolation, when combined with simultaneous image morphing, creates a wide range of facial expression changes [8].

Interpolated images are generated by varying the parameters of the interpolation functions. Geometric interpolation directly updates the 2D or 3D positions of the face mesh vertices, while parameter interpolation controls functions that indirectly move the vertices. For example, Sera et al. [9] perform a linear interpolation of the spring muscle force parameters, rather than the positions of the vertices, to achieve mouth animation.

Fig. 1.1. Linear Interpolation is performed on blend shapes. Left: Neutral pose, Right: "A" mouth shape, Middle: Interpolated shape.

Some recent efforts attempt to improve the efficiency of producing muscle actuation based blend shape animations [10, 11]. The Pose Space Deformation (PSD) method presented by Lewis et al. [12] provides a general framework for example-based interpolation which can be used for blendshape facial animations. In their work, the deformation of a surface (face) is treated as a function of some set of abstract parameters, such as {*smile, raise-eyebrow,...*}, and new surface is generated by scattered data interpolations.

Although interpolations are fast and they easily generate primitive facial animations, their ability to create a wide range of realistic facial configurations is restricted. Combinations of independent face motions are difficult to produce and non-orthogonal blend shapes often interfere each other, which cause animators have to go back and forth to re-adjust the weights of blend shapes. Lewis et al. [13] presented a user interface technique to automatically reduce blendshape interferences. Deng et al. [3] presented an automatic technique for mapping sparse facial motion capture data to pre-designed 3D blendshape face models by learning a radial basis functions based regression.

3 Parameterizations

Parameterization techniques for facial animation [14, 7, 15, 16] overcome some of the limitations and restrictions of simple interpolations. Ideal parameterizations specify any possible face and expression by a combination of independent parameter values [2]. Unlike interpolation techniques, parameterizations allow explicit control of specific facial configurations. Combinations of parameters provide a large range of facial expressions with relatively low computational costs.

As indicated in [17], there is no systematic way to arbitrate between two conflicting parameters to blend expressions that effect the same vertices. Thus, parameterization produces unnatural human expressions or configurations when a conflict between parameters occurs. For this reason, parameterizations

are designed to only affect specific facial regions. However, it often introduces noticeable motion boundaries. Another limitation of parameterization is that the choice of the parameter set depends on the facial mesh topology and, therefore, a complete generic parameterization is not possible. Furthermore, tedious manual tuning is required to set parameter values. The limitations of parameterization led to the development of diverse techniques such as morphing between images and geometry, physically faithful/pseudo muscle based animation, and performance driven animation.

4 Facial Action Coding System

The Facial Action Coding System (FACS) is a description of the movements of the facial muscles and jaw/tongue derived from an analysis of facial anatomy [18]. FACS includes forty four basic action units (AUs). Combinations of independent action units generate facial expressions. For example, combining the AU1 (Inner brow raiser), AU4 (Brow Raiser), AU15 (Lip Corner Depressor), and AU23 (Lip Tightener) creates a sad expression. A table of the sample action units and the basic expressions generated by the actions units are presented in Tables 1.1 and 1.2.

AU	FACS Name	AU	FACS Name	AU	FACS Name
1	Inner Brow Raiser	12	Lid Corner Puller	2	Outer Brow Raiser
14	Dimpler	4	Brow Lower	15	Lip Corner Depressor
5	Upper Lid Raiser	16	Lower Lip Depressor	6	Check Raiser
17	Chin Raiser	9	Nose Wrinkler	20	Lip Stretcher
23	Lip Tightener	10	Upper Lid Raiser	26	Jaw Drop

Table 1.1. Sample single facial action units.

Basic Expressions	Involved Action Units
Surprise	AU1, 2, 5, 15, 16, 20, 26
Fear	AU1, 2, 4, 5, 15, 20, 26
Anger	AU2, 4, 7, 9, 10, 20, 26
Happiness	AU1, 6, 12, 14
Sadness	AU1, 4, 15, 23

Table 1.2. Example sets of action units for basic expressions.

For its simplicity, FACS is widely utilized with muscle or simulated (pseudo) muscle based approaches. Animation methods using muscle models overcome the limitation of interpolation and provide a wide variety of facial

expressions. Physical muscle modeling mathematically describes the properties and the behavior of human skin, bone, and muscle systems. In contrast, pseudo muscle models mimic the dynamics of human tissue with heuristic geometric deformations. Despite its popularity, there are some drawbacks of using FACS [19]. First, AUs are purely local patterns while actual facial motion is rarely completely localized. Second, FACS offers spatial motion descriptions but not temporal components. In the temporal domain, co-articulation effects are lost in the FACS system.

5 Deformation Based Approaches

Direct deformation defined on the facial mesh surface often produces quality animation. It ignores underlying facial anatomy or true muscle structures. Instead, the focus is on creating various facial expressions by the manipulation of thin-shell mesh. This category includes morphing between different models and simulated pseudo muscles in the form of splines [20, 21, 22], wires [23], or free form deformations [24, 25].

5.1 2D and 3D Morphing

Morphing effects a metamorphosis between two target images or models. A 2D image morph consists of a warp between corresponding points in the target images and a simultaneous cross dissolve[3]. Typically, the correspondences are manually selected to suit the needs of the application. Morphs between carefully acquired and corresponded images produce very realistic facial animations. Beier and Neely [26] demonstrated 2D morphing between two images with manually specified corresponding features (line segments). The warp function is based upon a field of influence surrounding the corresponding features. Realism, with this approach, requires extensive manual interaction for color balancing, correspondence selection, and tuning of the warp and dissolve parameters. Variations in the target image viewpoints or features complicate the selection of correspondences. Realistic head motions are difficult to synthesize since target features become occluded or revealed during the animation.

To overcome the limitations of 2D morphs, Pighin et al. [27] combine 2D morphing with 3D transformations of a geometric model. They animate key facial expressions with 3D geometric interpolation while image morphing is performed between corresponding texture maps. This approach achieves viewpoint independent realism, however, animations are still limited to interpolations between pre-defined key facial expressions.

The 2D and 3D morphing methods can produce quality facial expressions, but they share similar limitations with the interpolation approaches. Selecting corresponding points in target images is manually intensive, dependent

[3] In cross dissolving, one image is faded out while another is simultaneously faded in.

on viewpoint, and not generalizable to different faces. Also, the animation viewpoint is constrained to approximately that of the target images.

5.2 Free From Deformation

Free form deformation (FFD) deforms volumetric objects by manipulating control points arranged in a three-dimensional cubic lattice [28]. Conceptually, a flexible object is embedded in an imaginary, clear, and flexible control box containing a 3D grid of control points. As the control box is squashed, bent, or twisted into arbitrary shapes, the embedded object deforms accordingly (Fig. 1.2). The basis for the control points is a trivariate tensor product Bernstein polynomial. FFDs can deform many types of surface primitives, including polygons; quadric, parametric, and implicit surfaces; and solid models.

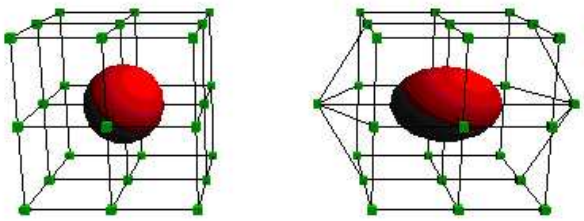

Fig. 1.2. Free form deformation. Controlling box and embedded object are shown. When controlling box is deformed by manipulating control points, so is embedded object.

Extended free form deformation (EFFD) [24] allows the extension of the control point lattice into a cylindrical structure. A cylindrical lattice provides additional flexibility for shape deformation compared to regular cubic lattices. Rational free form deformation (RFFD) incorporates weight factors for each control point, adding another degree of freedom in specifying deformations. Hence, deformations are possible by changing the weight factors instead of changing the control point positions. When all weights are equal to one, then RFFD becomes a FFD. A main advantage of using FFD (EFFD, RFFD) to abstract deformation control from that of the actual surface description is that the transition of form is no longer dependent on the specifics of the surface itself [29].

Displacing a control point is analogous to actuating a physically modeled muscle. Compared to Waters' physically based model [30], manipulating the positions or the weights of the control points is more intuitive and simpler than manipulating muscle vectors with delineated zone of influence. However, FFD (EFFD, RFFD) does not provide a precise simulation of the actual muscle and skin behavior. Furthermore, since FFD (EFFD, RFFD) is based upon

surface deformation, volumetric changes occurring in the physical muscle is not accounted for.

5.3 Spline Pseudo Muscles

Albeit polygonal models of the face are widely used, they often fail to adequately approximate the smoothness or flexibility of the human face. Fixed polygonal models do not deform smoothly in arbitrary regions, and planar vertices can not be twisted into curved surfaces without subdivision.

An ideal facial model has a surface representation that supports smooth and flexible deformations. Spline muscle models offer a plausible solution. Splines are usually up to C^2 continuous, hence a surface patch is guaranteed to be smooth, and they allow localized deformation on the surface. Furthermore, affine transformations are defined by the transformation of a small set of control points instead of all the vertices of the mesh reducing the computational complexity.

Some spline-based animation can be found in [20, 21]. Pixar used bicubic Catmull-Rom spline [4] patches to model Billy, the baby in animation *Tin Toy*, and used a variant of Catmull-Clark [31] subdivision surfaces to model *Geri*, a human character in short film Geri's game. This technique is mainly adapted to model sharp creases on a surface or discontinuities between surfaces [32]. For a detailed description of Catmull-Rom splines and Catmull-Clark subdivision surfaces, refer to [33, 31]. Eisert and Girod [34] used triangular B-splines to overcome the drawback that conventional B-splines do not refine curved areas locally since they are defined on a rectangular topology.

A hierarchical spline model reduces the number of unnecessary control points. Wang et al. [22] showed a system that integrated hierarchical spline models with simulated muscles based on local surface deformations. Bicubic B-splines offer both smoothness and flexibility, which are hard to achieve with conventional polygonal models. The drawback of using naive B-splines for complex surfaces becomes clear, however, when a deformation is required to be finer than the patch resolution. To produce finer patch resolution, an entire row or column of the surface is subdivided. Thus, more detail (and control points) is added where none are needed. In contrast, hierarchical splines provide the local refinements of B-spline surfaces and new patches are only added within a specified region. Hierarchical B-splines are an economical and compact way to represent a spline surface and achieve high rendering speed. Muscles coupled with hierarchical spline surfaces are capable of creating bulging skin surfaces and a variety of facial expressions.

[4] A distinguishing property of Catmull-Rom splines is that the piecewise cubic polynomial segments pass through all the control points except the first and last when used for interpolation. Another is that the convex hull property is not observed in Catmull-Rom spline.

6 Physics Based Muscle Modeling

Physics-based muscle models fall into three categories: mass spring systems, vector representations, and layered spring meshes. Mass-spring methods propagate muscle forces in an elastic spring mesh that models skin deformation. The vector approach deforms a facial mesh using motion fields in delineated regions of influence. A layered spring mesh extends a mass spring structure into three connected mesh layers to model anatomical facial behavior more faithfully.

6.1 Spring Mesh Muscle

The work by Platt and Badler [35] is a forerunner of the research focused on muscle modeling and the structure of the human face. Forces applied to elastic meshes through muscle arcs generate various facial expressions. Platt's later work [36] presents a facial model with muscles represented as collections of functional blocks in defined regions of the facial structure. Platt's model consists of thirty eight regional muscle blocks interconnected by a spring network. Action units are created by applying muscle forces to deform the spring network. There are some recent developments using mass-spring muscles for facial animation [37, 38]. For example, Kahler et al. [38] present a convenient editing tool to interactively specify mass-spring muscles into 3D face geometry.

6.2 Vector Muscle

A very successful muscle model was proposed by Waters [30]. A delineated deformation field models the action of muscles upon skin. A muscle definition includes the vector field direction, an origin, and an insertion point (the left panel of Figure 1.3). The field extent is defined by cosine functions and fall off factors that produce a cone shape when visualized as a height field. Waters also models the mouth sphincter muscles as a simplified parametric ellipsoid. The sphincter muscle contracts around the center of the ellipsoid and is primarily responsible for the deformation of the mouth region. Waters animates human emotions such as anger, fear, surprise, disgust, joy, and happiness using vector based linear and orbicularis oris muscles utilizing the FACS. The right panel of Figure 1.3 shows the Waters' muscles embedded in a facial mesh.

The positioning of vector muscles into anatomically correct positions can be a daunting task. The process involves manual trial and error with no guarantee of efficient or optimal placement. Incorrect placement results in unnatural or undesirable animation of the mesh. Nevertheless, the vector muscle model is widely used because of its compact representation and independence of the facial mesh structure. An example of vector muscles is seen in Billy, the baby in the movie *Tin Toy*, who has forty seven Waters' muscles on his face.

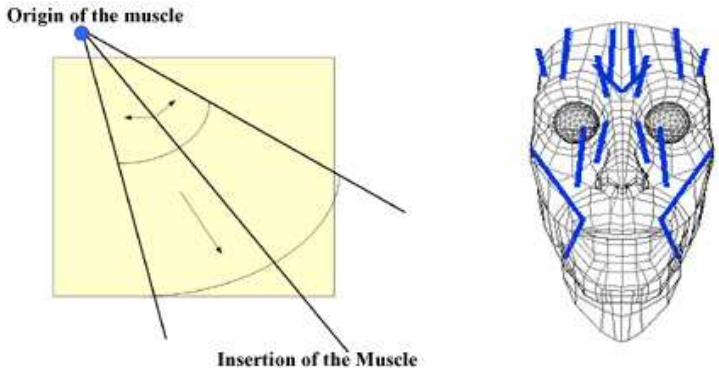

Fig. 1.3. The left panel shows the zone of influence of Waters' linear muscle model. The right panel shows muscle placement in Waters's work [30].

6.3 Layered Spring Mesh Muscles

Terzopoulos and Waters [39] proposed a facial model that models detailed anatomical structure and dynamics of the human face. Their three-layers of deformable mesh correspond to skin, fatty tissue, and muscle tied to bone. Elastic spring elements connect each mesh node and each layer. Muscle forces propagate through the mesh systems to create animation. This model is faithful to facial anatomy. Simulating volumetric deformations with three-dimensional lattices, however, requires extensive computation. A simplified mesh system reduces the computation time while still maintaining similar quality [40].

Lee et al. [41, 42] presented face models composed of physics-based synthetic skin and muscle layers based on earlier work [39]. The face model consists of three components: a biological tissue layer with nonlinear deformation properties, a muscle layer knit together under the skin, and an impenetrable skull structure beneath the muscle layer. The synthetic tissue is modeled as triangular prism elements that are divided into the epidermal surface, the fascia surface, and the skull surface. Spring elements connecting the epidermal and fascia layers simulate skin elasticity. Spring elements that effect muscle forces connect the fascia and skull layers. The model achieves better fidelity. Tremendous computation is required, however, and extensive tuning is needed to model a specific face or characteristic.

7 3D Face Modeling

An important problem in facial animation is to model a specific person, i.e., modeling the 3D geometry of an individual face. A range scanner, digitizer probe, or stereo disparity can measure three-dimensional coordinates. The

models obtained by those processes are often poorly suited for facial animation. Information about the facial structures is missing; measurement noise produces distracting artifacts; and model vertices are poorly distributed. Also, many measurement methods produce incomplete models, lacking hair, ears, eyes, etc. Therefore, post processing on the measured data is often necessary.

7.1 Person Specific Model Creation

An approach to person-specific modeling is to painstakingly prepare a generic animation mesh with all the necessary structure and animation information. This generic model is fitted or deformed to a measured geometric mesh of a specific person to create a personalized animation model. The geometric fit also facilitates the transfer of texture if it is captured with the measured mesh. If the generic model has fewer polygons than the measured mesh, decimation is implicit in the fitting process.

Person-specific modeling and fitting processes use various approaches such as scattered data interpolations [43, 5, 44] and projections onto the cylindrical coordinates incorporated with a positive Laplacian field function [42]. Some methods attempt an automated fitting process, but most require manual intervention.

Radial basis functions are capable of closely approximate or interpolate smooth hypersurfaces [45] such as human facial shapes. Some approaches morph a generic mesh into specific shapes with scattered data interpolation techniques based on radial basis functions. The advantages of this approach are as follows. First, the morph does not require equal numbers of nodes in the involved meshes since missing points are interpolated [43]. Second, mathematical support ensures that a morphed mesh approaches the target mesh, if appropriate correspondences are selected [45, 46].

A typical process of 3D volume morphing is as follows. First, biologically meaningful landmark points are manually selected around the eyes, nose, lips, and perimeters of both face models. Second, the landmark points define the coefficients of the kernel of the radial basis function used to morph the volume. Finally, points in the generic mesh are interpolated using the coefficients computed from the landmark points. The success of the morphing depends strongly on the selection of the landmark points [43, 27].

Instead of morphing a face model, a morphable model exploits a preconstructed set of face database to create a person specific model [47]. First, a scanning process collects a large number of faces to compile a database. This example 3D face models spans the space of any possible human faces in terms of geometry and texture. New faces and expressions can be represented as a linear combination of the examples. Typically, an image of a new person is provided to the system, then, the system outputs a 3D model of the person that closely matches the image.

7.2 Anthropometry

The generation of individual models using anthropometry [5] attempts to produce facial variations where absolute appearance is not important. Kuo et al. [48] proposes a method to synthesize a lateral face from one 2D gray-level image of a frontal face. A database is first constructed, containing facial parameters measured according to anthropomorphic definitions. This database serves as a priori knowledge. The lateral facial parameters are estimated from frontal facial parameters by using minimum mean square error (MMSE) estimation rules applied to the database. Specifically, the depth of one lateral facial parameter is determined by the linear combination of several frontal facial parameters. The 3D generic facial model is then adapted according to both the frontal plane coordinates extracted from the image and their estimated depths. Finally, the lateral face is synthesized from the feature data and texture-mapped.

DeCarlo et al. [49] constructs various facial models purely based on anthropometry without assistance from images. This system constructs a new face model in two steps. The first step generates a random set of measurements that characterize the face. The form and values of these measurements are computed according to face anthropometry (Figure 1.4). The second step constructs the best surface that satisfies the geometric constraints using a variational constrained optimization technique [50, 51]. In this technique, one imposes a variety of constraints on the surface and then tries to create a smooth and fair surface while minimizing the deviation from a specified rest shape, subject to the constrains. For a face modeling, anthropometric measurements are the constraints, and the remainder of the face is determined by minimizing the deviation from the given surface objective function. Variational modeling enables the system to capture the shape similarities of faces, while allowing anthropometric differences. Although anthropometry has potential for rapidly generating plausible facial geometric variations, the approach does not model realistic variations in color, wrinkling, expressions, or hair.

8 Performance Driven Facial Animation

The difficulties in controlling facial animations led to the performance driven approach where tracked human actors drive the animation. Real time video processing allows interactive animations where the actors observe the animations they create with their motions and expressions. Accurate tracking of feature points or edges is important to maintain a consistent and quality of animation. Often the tracked 2D or 3D feature motions are filtered or transformed to generate the motion data needed for driving a specific animation system. Motion data can be used to directly generate facial animation [19]

[5] the science dedicated to the measurements of the human face

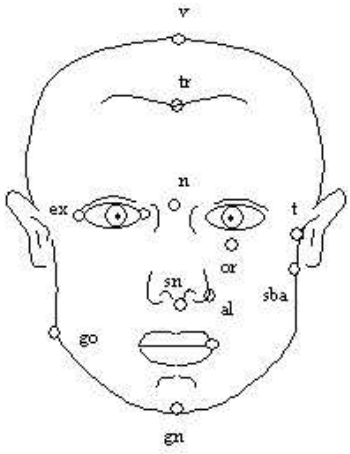

Fig. 1.4. Some of the anthropometric landmarks on the face. The selected landmarks are widely used as measurements for describing the human face.

or to infer AUs of FACS in generating facial expressions. Figure 1.5 shows animation driven from a real time feature tracking system.

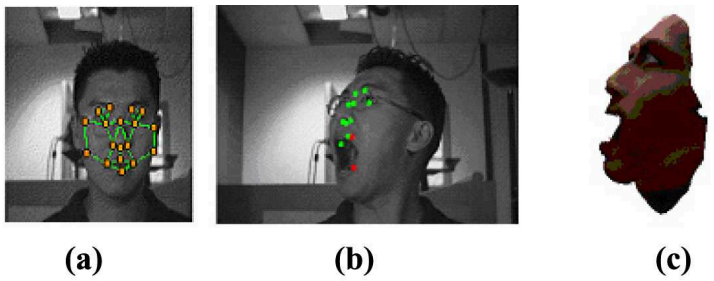

Fig. 1.5. Real time tracking is performed without markups on the face using Eyematic Inc.'s face tracking system. Real time animation of the synthesized avatar is achieved based on the 11 tracked features. Here (a) shows the initial tracking of the face features, (b) shows features are tracked in real time while the subject is moving, and (c) shows an avatar mimics the behavior of the subject.

8.1 Snakes and Markings

Snakes, or deformable minimum-energy curves, are used to track intentionally marked facial features [52]. The recognition of facial features with snakes is primarily based on color samples and edge detection. Many systems couple

tracked snakes to underlying muscles mechanisms to drive facial animation [53, 54, 39, 55, 56]. Muscle contraction parameters are estimated from the tracked facial displacements in video sequences.

Tracking errors accumulate over long image sequences. Consequently, a snake may lose the contour it is attempting to track. In [57], tracking from frame to frame is done for the features that are relatively easy to track. A reliability test enables a reinitialization of a snake when error accumulations occur.

8.2 Optical Flow Tracking

Colored markers painted on the face or lips [58, 59, 60, 61, 62, 63, 64, 9] are extensively used to aid in tracking facial expressions or recognizing speech from video sequences. However, markings on the face are intrusive. Also, reliance on markings restricts the scope of acquired geometric information to the marked features. Optical flow [65] and spatio-temporal normalized correlation measurements [66] perform natural feature tracking and therefore obviate the need for intentional markings on the face [67, 19]. Chai et al. [68] propose a data-driven technique to translate noisy, low-quality 2D tracking signals from video to high-quality 3D facial animations based on a pre-processed facial motion database. One limitation of this approach is that a pre-processed facial motion database is required, and its performance may depend on the match between pre-recorded persons in the database and target face models. Zhang et al. [69] propose a space-time stereo tracking algorithm to build 3D face models from video sequences that maintain point correspondences across the entire sequence without using any marker.

8.3 Facial Motion Capture Data

More recent trend to produce quality animation is to use 3D motion capture data. Motion capture data have successfully been used in recent movies such as *Polar Express* and *Monster House*. Typically, motion data is captured and filtered prior to the animation. An array of high performance cameras is utilized to reconstruct the 3D maker locations on the face. Although this optical system is difficult to set up and expensive, the reconstructed data provide accurate timing and motion information. Once the data is available, facial animation can be created by employing underlying muscle structure [70] or Blendshapes [71, 72, 3].

9 MPEG-4 Facial Animation

Due to its increased applications, facial animation was adopted into the MPEG-4 standard, an object-based multimedia compression standard [73].

MPEG-4 specifies and animates 3D face models by defining Face Definition Parameters (FDP) and Facial Animation Parameters (FAP). FDPs enclose information for constructing specific 3D face geometry, and FAPs encode motion parameters of key feature points on the face over time. Face Animation Parameter Units (FAPU) that scale FAPs for fitting any face model, are defined as the fractions of key facial features, such as the distance between the two eyes.

In MPEG-4 facial animation standard, total 84 Feature Points (FPs) are specified. Figure 1.6 approximately illustrates part of the MPEG-4 feature points in a front face. After excluding the feature points that are not affected by FAPs, 68 FAPs are categorized into groups (Table 1.3). Most of FAP groups are low-level parameters since they precisely specify how much a given FP should be moved. One FAP group (visemes and expressions) is considered as high-level parameters, because these parameters are not precisely specified. For example, textual descriptions are used to describe expressions. As such, reconstructed facial animation depends on the implementation of individual MPEG-4 facial animation decoder programs.

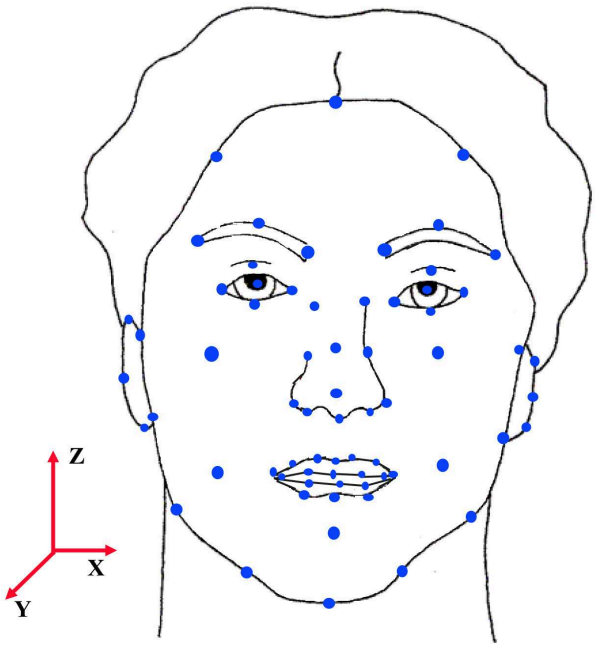

Fig. 1.6. Part of facial feature points defined in the MPEG-4 standard. A complete description of the MPEG-4 feature points can be found in [74].

Group	Number of FAPs
Viseme and expressions	2
Lip, Chin and Jaw	26
Eyes (including pupils and eyelids)	12
Eyebrow	8
Cheeks	4
Tongue	5
Head Movement	3
Nose	4
Ears	4

Table 1.3. FAP groups in MPEG-4.

Previous research efforts on MPEG-4 facial animation were focused on deforming 3D face models based on MPEG-4 feature points [75, 76] and building MPEG-4 facial animation decoder systems [77, 78, 79, 80]. For example, Escher et al. [75] deform a generic face model using a free-form deformation based approach to generate MPEG-4 facial animations. Kshirsagar et al. [76] propose an efficient feature-point based face deformation technique given MPEG-4 feature point inputs. In their approach, the motion of each MPEG-4 feature point is propagated to neighboring vertices of the face model and the motion of each vertex (non-feature point) is the summation of these motion propagations. Various MPEG-4 facial animation decoder systems [77, 78] and frameworks that are targeted for web and mobile applications [79, 80] are also proposed. For more details of MPEG-4 facial animation standard, implementations and applications, please refer to the MPEG-4 facial animation book [81].

10 Visual Speech Animation

Visual speech animation can be regarded as visual motions of the face (especially the mouth part) when humans are speaking. Synthesizing realistic visual speech animations corresponding to novel text or pre-recorded acoustic speech input has been a difficult task for decades, because human languages, such as English, generally have not only a large vocabulary and a large number of phonemes (the theoretical representation of an utterance/sound), but also the phenomena of *speech co-articulation* that complicates the mappings between acoustic speech signals (or phonemes) and visual speech motions. In linguistics literature, speech co-articulation is defined as follows: phonemes are not pronounced as an independent sequence of sounds, but rather that the sound of a particular phoneme is affected by adjacent phonemes. Visual speech co-articulation is analogous.

Previous research efforts in visual speech animation generation can be roughly classified into two different categories: viseme-driven approaches and data-driven approaches. Viseme-driven approaches require animators to de-

sign key mouth shapes for phonemes (termed as *visemes*) in order to generate novel speech animations. On the contrary, data-driven approaches do not need pre-designed key shapes, but generally require a pre-recorded facial motion database for synthesis purposes.

10.1 Viseme-Driven Approaches

Viseme is defined as a basic visual unit that corresponds to the phoneme in speech. Viseme-driven approaches typically require animators to design visemes (key mouth shapes), and then empirical smooth functions [82, 83, 14, 84, 85, 86] or co-articulation rules [87, 88, 89] are used to synthesize novel speech animations.

Given novel speech sound track and a small number of visemes, J.P. Lewis [83] proposes an efficient lip-sync technique based on a linear prediction model. Cohen and Massaro [14] propose the Cohen-Massaro co-articulation model for generating speech animations. In their approach, a viseme shape is defined via dominance functions that are defined in terms of each facial measurement, such as such as lips, tongue tip, etc. And the weighted sum of dominance values determines final mouth shapes. Figure 1.7 schematically illustrates the essential idea of the Cohen-Massaro Model. Its recent extensions [84, 85, 86, 89] further improved the Cohen-Massaro co-articulation model. For example, Cosi et al. [85] added a temporal resistance function and a shape function for more general cases, such as fast/slow speaking rates. The approach proposed by Goff and Benoît [84] calculates the model parameter values of the Cohen-Massaro model by analyzing parameter trajectories measured from a French speaker. The approach proposed by King and Parent [86] extends the Cohen-Massaro model by using viseme curves to replace a single viseme target. Bevacqua and Pelachaud [89] propose an expressive qualifier modeled from recorded speech motion data to make expressive speech animations.

Rule-based co-articulation models [87, 88] leave some visemes undefined based on their co-articulation importance and phoneme contexts. These approaches are based on an important observation that phonemes have different sensitivity to their phoneme context: some phonemes (and their visemes) are strongly affected by neighboring phonemes (and visemes), while some others are less affected. Deng et al. [90, 91, 92] propose a novel motion capture mining technique that "learns" speech co-articulation models for diphones (a phoneme pair) and triphones from the pre-recorded facial motion data, and then generates novel speech animations by blending pre-designed visemes (key mouth shapes) using the learned co-articulation models.

Animation realism generated by the above viseme-driven approaches largely depends on the hand-crafted smoothing (co-articulation) functions and a hidden assumption that a viseme can be represented by one or several pre-designed key shapes. However, in practice, constructing accurate co-articulation functions and phoneme-viseme mappings requires challenging and

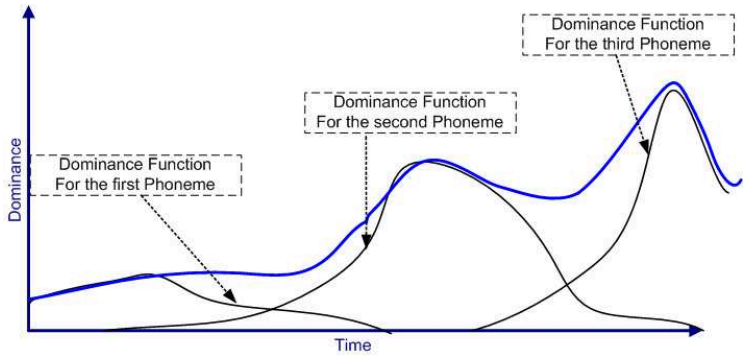

Fig. 1.7. Schematic illustration of the Cohen-Massaro co-articulation model [14]. Dominance functions of three consecutive phonemes are plotted, and weighted sum of dominance curves is plotted as a blue curve.

painstaking manual work. As a new trend for speech animation generation, data-driven approaches were proposed to alleviate the painstaking manual work.

10.2 Data-Driven Approaches

Data-driven approaches synthesize new speech animations by concatenating pre-recorded facial motion data or sampling from statistical models learned from the data. Their general pipeline is as follows. First, facial motion data (2D facial images or 3D facial motion capture data) are pre-recorded. Second, there are two different ways to deal with the constructed facial motion database: either statistical models for facial motion control are trained from the data (learning-based approaches), or the facial motion database is further organized and processed (sample-based approaches). Finally, given novel sound track or text input, corresponding visual speech animations are generated by sampling from the trained statistical models, or recombining motion frames optimally chosen from the facial motion database. Figure 1.8 shows a schematic view of the data-driven speech animation approaches.

The data-driven approaches typically generate realistic speech animation results, but it is hard to predict how much motion data are enough to train statistical models or construct a balanced facial motion database. In other words, the connection from the amount of pre-recorded facial motion data to the realism of synthesized speech animations is not clear. Furthermore, these approaches often do not provide intuitive process controls for the animators.

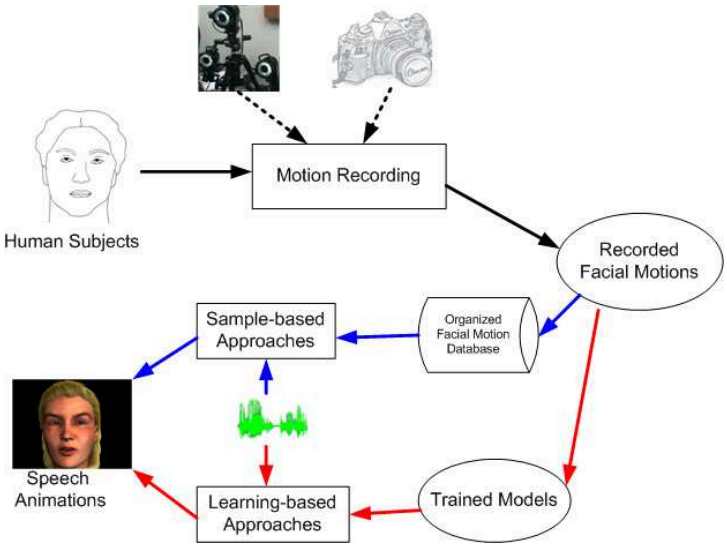

Fig. 1.8. Sketched general pipeline of data-driven speech animation generation approaches. The sample-based approaches go with the blue path, and the learning-based approaches go with the red path.

Sample-Based

Bregler et al. [93] present the "video rewrite" method for synthesizing 2D talking faces given novel speech input, based on the collected "triphone video segments". Instead of using ad hoc co-articulation models and ignoring dynamics factors in speech, this approach models the co-articulation effect with "triphone video segments", but it is not generative (i.e. the co-articulation cannot be applied to other faces without retraining). The work of [94, 95] further extends "the triphone combination idea" [93] to longer phoneme segments. For example, Cao et al. [95, 96] propose a greedy search algorithm to look for longer pre-recorded facial motion sequences ($\geq$ 3 phonemes) in the database. The work of [97, 98, 99, 100] searches for the optimal combination of pre-recorded motion frame sequences by introducing various cost functions, based on dynamic programming based search algorithms. In the work of [100], a phoneme-Isomap interface is introduced to provide high-level controls for the animators, and phoneme-level emotion specifiers are enforced as search constraints.

Instead of constructing phoneme segment database [93, 97, 94, 95, 98, 99, 100, 101], Kshirsagar and Thalmann [102] propose a syllable motion based approach to synthesize novel speech animations. In their approach, captured facial motions are segmented into syllable motions, and then new speech animations are achieved by concatenating syllable motion segments optimally

chosen from the syllable motion database. Sifakis et al. [103] propose a physics-based approach to generate novel speech animations by first computing muscle activation signals for each phoneme (termed as *physemes*) enclosed in the pre-recorded facial motion data and then concatenating corresponding physemes given novel speech input.

Learning-Based

Learning-based approaches model speech co-articulations as implicit functions in statistical models. Brand [104] learns a HMM-based facial control model by an entropy minimization learning algorithm from voice and video training data and then effectively synthesizes full facial motions for novel audio track. This approach models co-articulations, using the Viterbi algorithm through vocal HMMs to search for most likely facial state sequence that is used for predicting facial configuration sequences. Ezzat et al. [105] learn a multidimensional morphable model from a recorded face video database that requires a limited set of mouth image prototypes and use the magnitude of diagonal covariance matrices of phoneme clusters to represent co-articulation effects: the larger covariance of a phoneme cluster means this phoneme has a smaller co-articulation, and vice versa.

Blanz et al. [106] reanimate 2D faces in images and video by reconstructing 3D face model using the morphable face model framework [47] and learning an expression and viseme space from scanned 3D faces. This approach addresses both speech and expressions. Deng et al. [91, 92] proposed an expressive speech animation system that learn speech co-articulation models and expression eigen-spaces from recorded facial motion capture data. Some other approaches [107, 108] were also proposed for generating expressive speech animations.

Generally, these approaches construct economical and compact representations for human facial motions and synthesize human-like facial motions. However, how much data are minimally required to guarantee satisfied synthesis results is an unsolved issue in these approaches, and creating explicit correlations between training data and the realism of final animations would be a critical need. Furthermore, model and feature selections residing in many machine learning algorithms are still far away from being solved.

11 Facial Animation Editing

Editing facial animations by posing key faces is a widely-used practice. Instead of moving individual vertex of 3D face geometry, various deformation approaches (Section 5) and the blendshape methods (Section 2) can be regarded to simultaneously move and edit a group of relevant vertices, which greatly improve the efficiency of facial animation editing. However, different facial regions are essentially correlated each other, and the above deformation

approaches typically operate a local facial region at one time. The animators need to switch editing operations on different facial regions in order to sculpt 3D realistic faces with fine details, which creates a large amount of additional work for the animators. In addition, even for skilled animators, it is difficult to judge which facial pose (configuration) is closer to a real human face. Some recent work in facial animation editing [109, 72, 13, 110, 111, 112] were proposed to address this issue.

The ICA-based facial motion editing technique [109] applies Independent Component Analysis (ICA) onto pre-recorded expressive facial motion capture data and interprets certain ICA components as expression and speech-related components. Further editing operations, e.g. scaling, are performed on these ICA components in their approach. Chang and Jenkins [112] propose a 2D sketch interface for posing 3D faces. In their work, users can intuitively draw 2D strokes in 2D face space that are used to search for the optimal pose of the face.

Editing a local facial region while preserving naturalness of the whole face is another intriguing idea. The geometry-driven editing technique [110] generates expression details on 2D face images by constructing a PCA-based hierarchical face representation from a selected number of training 2D face images. When users move one or several points on the 2D face image, the movements of other facial control points are automatically computed by a motion propagation algorithm. Based on a blendshape representation for 3D face models, Joshi et al. [72] propose an interactive tool to edit 3D face geometry by learning controls through a physically-motivated face segmentation. A rendering algorithm for preserving visual realism in this editing was also proposed in their approach.

Besides the above approaches, the morphable face model framework [47] and the multilinear face model [111] can be used for facial animation editing: once these statistical models are constructed from training face data, users can manipulate high-level attributes of the face, such as gender and expression, to achieve the purpose of facial animation editing.

12 Facial Animation Transferring

Automatically transferring facial motions from an existing (source) model to a new (target) model can significantly save painstaking and model-specific animation tuning for the new face model. The source facial motions can have various formats, including 2D video faces, 3D facial motion capture data, and animated face meshes, while the target models typically are a static 3D face mesh or a blendshape face model. In this regard, performance driven facial animation described in Section 8 can be conceptually regarded as one specific way of transferring facial motions from 2D video faces to 3D face models. In this section, we will review other facial animation transferring techniques.

Transferring facial motions between two 3D face meshes can be performed through geometric deformations. Noh and Neumann [113] propose an "expression cloning" technique to transfer vertex displacements from a source 3D face model to target 3D face models that may have different geometric proportions and mesh structure. Its basic idea is to construct vertex motion mappings between models through the Radial Basis Functions (RBF) morphing. Sumner and Popović [114] propose a general framework that automatically transfer geometric deformations between two triangle meshes, which can be directly applied to retarget facial motions from one source face mesh to a target face mesh. Both approaches need a number of initial face landmark correspondences through either heuristic rules [113] or manually specifying.

A number of approaches were proposed to transfer source facial motions to blendshape face models [70, 115, 98, 11, 3] due to the popularized use of blendshape methods in industry practice. Choe and Ko [70] transfer tracked facial motions to target blendshape face models composed of hand-generated muscle actuation base, by iteratively adjusting muscle actuation base and analyzed weights through an optimization procedure. The work of [115, 98] transfers facial animations using example-based approaches. Essentially these approaches require animators to sculpt proper blendshape face models based on a set of key facial poses, delicately chosen from source facial animation sequences. Hence, it is difficult to apply these techniques to pre-designed blendshape models without considerable efforts. Sifakis et al. [11] first create an anatomically accurate face model composed of facial musculature, passive tissue, and underlying skeleton structure, and then use nonlinear finite element methods to determine accurate muscle actuations from the motions of sparse facial markers. Anatomically accurate 3D face models are needed for this approach, which is another challenging task itself in computer animation. Deng et al. [3] propose an automatic technique to directly map 3D facial motion capture data to pre-designed blendshape face models. In their approach, Radial Basis Functions (RBF) networks are trained to map a new motion capture frame to its corresponding blendshape weights, based on chosen training pairs between mocap frames and blendshape weights.

The above approaches trustily "copy" facial motions between models, but they provide little transformation function, for example, change affective mode during transferring. Bilinear models and multilinear models were proposed to transform facial motions [116, 117, 111]. Chuang and Bregler [116, 117] learn a facial expression mapping/transformation function from training video footage using bilinear models [118], and then this learned mapping is used to transform input video of neutral talking to expressive talking. Vlasic et al. [111] propose a framework to transfer facial motion in video to other 2D or 3D faces by learning statistical multilinear models from scanned 3D face meshes. In their work, the learned multilinear models are controlled via intuitive attribute parameters, such as identity and expression. Varying one attribute parameter (e.g. identity) while keeping other attributes intact, can transfer the facial motions from one model to another. Both approaches inter-

pret expressions as dynamic processes, but the expressive face frames retain the same timing as the original neutral speech, which does not seem plausible in all cases.

13 Facial Gesture Generation

Facial gesture is typically interpreted as a gesture executed with the facial muscles and facial movement, enclosing various visual components, such as facial expressions, head movement, etc. In this section we focus on reviewing previous research efforts in eye motion synthesis and head movement generation. As for generating facial expressions on virtual characters, refer to the state of the art report written by Vinayagamoorthy et al. [119].

As "windows to the soul", the eyes are particularly scrutinized and subtle, since eye gaze is one of the strongest cues to the mental state of human beings when someone is talking, they look to our eyes to judge our interest and attentiveness, and we look into their eyes to signal our intent to talk. Chopra-Khullar et al. [120] propose a framework for computing gestures including eye gaze and head motions of virtual agents in dynamic environments, given high-level scripts. Vertegaal et al. [121, 122] studied whether eye gaze direction clues can be used as a reliable signal for determining who is talking to whom in multi-party conversations. Lee et al. [123] treat "textural" aspects of gaze movement using statistical approaches, and demonstrated the necessity of the gaze details for achieving realism and conveying an appropriate mental state. In their approach, signals from an eye tracker are analyzed to produce a statistical model of eye saccades. However, only first-order statistics are used, and gaze-eyelid coupling and vergence are not considered in their work. Deng et al. [124, 125] propose a texture synthesis based technique to simultaneously synthesize realistic eye gaze and blink motion, accounting for any possible correlations between the two.

Natural head motion is an indispensable part of realistic facial animation and engaging human computer interface. A number of approaches were proposed to generate head motions for talking avatars [128, 129, 130, 131, 132, 133, 126, 127]. Rule-based approaches [128, 129] generate head motions from labeled text by pre-defined rules, but their focus was only the "nodding". Graf et al. [130] estimated the conditional probability distribution of major head movements (e.g. nodding) given the occurrences of pitch accents, based on their collected head motion data. Chuang and Bregler [132] generate head motions corresponding to novel acoustic speech input, by combining best-matched recorded head motion segments in the constructed pitch-indexed database. Deng et al. [133] synthesize appropriate head motions with keyframing controls, where a constrained dynamic programming algorithm was used to generate an optimal head motion sequence that maximally satisfies both acoustic speech and key frame constraints (e.g. specified key head poses). Busso et al. [126] presented a Hidden Markov Models (HMMs) based

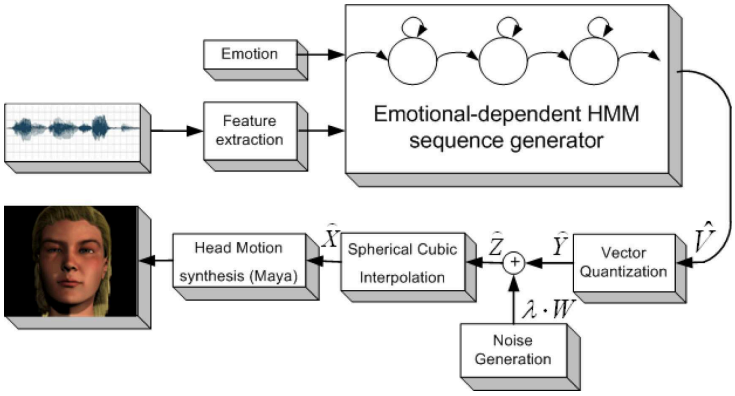

Fig. 1.9. Schematic overview of the HMM-based expressive head motion synthesis framework [126, 127].

framework to generate natural head motions directly from acoustic prosodic features. This framework was further extended to generate expressive head motions [127]. Figure 1.9 shows a schematic overview of the HMM-based head motion synthesis framework [127].

14 Summary

We surveyed various computer facial animation techniques and classified them into the following categories: blendshape method (shape interpolation), parameterizations, Facial Action Coding Systems based approaches, deformation based approaches, physics based muscle modeling, 3D face modeling, performance driven facial animation, MPEG-4 facial animation, visual speech animation, facial animation editing, facial animation transferring, and facial gesture generation. Within each category, we described main ideas of its approaches and compare their strength and weakness.

References

1. F. Parke. Computer generated animation of faces. In *Proc. ACM Nat'l Conf.*, volume 1, pages 451–457, 1972.
2. F.I. Parke and K. Waters. *Computer Facial Animation*. 1996.
3. Z. Deng, P.Y. Chiang, P. Fox, and U. Neumann. Animating blendshape faces by cross mapping motion capture data. In *Proceeding of ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 2006.
4. P. Bergeron and P. Lachapelle. Controlling facial expression and body movements in the computer generated short "tony de peltrie", 1985.

5. F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D.H. Salesin. Synthesizing realistic facial expressions from photographs. In *SIGGRAPH Proceedings*, pages 75–84, 1998.
6. K. Waters and T.M. Levergood. Decface: An automatic lip-synchronization algorithm for synthetic faces, 1993.
7. F.I. Parke. A parametric model for human faces. *Ph.D. Thesis, University of Utah, UTEC-CSc-75-047*, 1974.
8. K. Arai, T. Kurihara, and K. Anjyo. Bilinear interpolation for facial expression and metamorphosis in real-time animation. *The Visual Computer*, 12:105–116, 1996.
9. H. Sera, S. Morishma, and D. Terzopoulos. Physics-based muscle model for moth shape control. In *IEEE International Workshop on Robot and Human Communication*, pages 207–212, 1996.
10. B. W. Choe and H. S. Ko. Analysis and synthesis of facial expressions with hand-generated muscle actuation basis. In *IEEE Computer Animation Conference*, pages 12–19, 2001.
11. E. Sifakis, I. Neverov, and R. Fedkiw. Automatic determination of facial muscle activations from sparse motion capture marker data. *ACM Trans. Graph.*, 24(3):417–425, 2005.
12. J.P. Lewis, M. Cordner, and N. Fong. Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation. In *SIGGRAPH proceedings*, pages 165–172, 2000.
13. J.P. Lewis, J. Mooser, Z. Deng, and U. Neumann. Reducing blendshape interference by selected motion attenuation. In *Proceedings of ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3DG)*, pages 25–29, 2005.
14. M. Cohen and D. Massaro. Modeling co-articulation in synthetic visual speech. *Model and Technique in Computer Animation*, pages 139–156, 1993.
15. F.I. Parke. Parameterized models for facial animation. *IEEE Computer Graphics and Applications*, 2(9):61–68, 1982.
16. F.I. Parke. Parameterized models for facial animation revisited, 1989.
17. K. Waters and J. Frisbie. A coordinated muscle model for speech animation. In *Graphics Interface'95*, pages 163–170, 1995.
18. P. Ekman and W.V. Friesen. *Facial Action Coding System*. Consulting Psychologists Press, 1978.
19. I.A. Essa, S. Basu, T. Darrell, and A. Pentland. Modeling, tracking and interactive animation of faces and heads using input from video. In *Proceedings of Computer Animation*, pages 85–94, 1996.
20. M. Nahas, H. Hutric, M. Rioux, and J. Domey. Facial image synthesis using skin texture recording. *Visual Computer*, 6(6):337–343, 1990.
21. M.L. Viad and H. Yahia. Facial animation with wrinkles. In *Proceedings of the Third Eurographics Workshop on Animation and Simulation*, 1992.
22. CLY Wang and D.R. Forsey. Langwidere: A new facial animation system. In *Proceedings of Computer Animation*, pages 59–68, 1994.
23. K. Singh K and E. Fiume. Wires: A geometric deformation technique. In *SIGGRAPH Proceedings*, pages 405–414, 1998.
24. S. Coquillart. Extended free-form deformation: A sculpturing tool for 3d geometric modeling. *Computer Graphics*, 24:187–193, 1990.
25. P. Kalra, A. Mangili, N.M. Thalmann, and D. Thalmann. Simulation of facial muscle actions based on rational free from deformations. In *Eurographics*, volume 11, pages 59–69, 1992.

26. T. Beier and S. Neely. Feature-based image metamorphosis. In *SIGGRAPH proceedings*, pages 35–42. ACM Press, 1992.
27. F. Pighin, J. Auslander, D. Lischinski, D.H. Salesin, and R. Szeliski. Realistic facial animation using image-based 3d morphing, 1997.
28. T.W. Sederberg and S.R. Parry. Free-form deformation of solid geometry models. In *Computer Graphics, SIGGRAPH*, volume 20, pages 151–160, 1996.
29. N. M. Thalmann and D. Thalmann. *Interactive Computer Animation*. Prentice Hall, 1996.
30. K. Waters. A muscle model for animating three-dimensional facial expression. In *SIGGRAPH Proceedings*, volume 21, pages 17–24, 1987.
31. E. Catmull and J. Clark. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer Aided Design*, 10(6):350–355, 1978.
32. T. Derose, M. Kass, and T. Truong. Subdivision surfaces in character animation. In *SIGGRAPH Proceedings*, pages 85–94, 1998.
33. E. Catmull. Subdivision algorithm for the display of curved surfaces. *Ph.D. Thesis, University of Utah*, 1974.
34. P. Eisert and B. Girod. Analyzing facial expressions for virtual conferencing. *IEEE Computer Graphics and Applications*, 18(5):70–78, 1998.
35. S. Platt and N. Badler. Animating facial expression. computer graphics. *Computer Graphics*, 15(3):245–252, 1981.
36. S.M. Platt. A structural model of the human face. *Ph.D. Thesis, University of Pennsylvania*, 1985.
37. Y. Zhang, E. C. Parkash, and E. Sung. A physically-based model with adaptive refinement for facial animation. In *Proc. of IEEE Computer Animation'2001*, pages 28–39, 2001.
38. K. Kähler, J. Haber, and H. P. Seidel. Geometry-based muscle modeling for facial animation. In *Proc. of Graphics Interface'2001*, 2001.
39. D. Terzopoulos and K. Waters. Physically-based facial modeling, analysis, and animation. *Journal of Visualization and Computer Animation*, 1(4):73–80, 1990.
40. Y. Wu, N.M. Thalmann, and D. Thalmann. A plastic-visco-elastic model for wrinkles in facial animation and skin aging. In *Proc. 2nd Pacific Conference on Computer Graphics and Applications, Pacific Graphics*, 1994.
41. Y. Lee, D. Terzopoulos, and K. Waters. Constructing physics-based facial models of individuals. In *Proc. of Graphics Interface'93*, 1993.
42. Y.C. Lee, D. Terzopoulos, and K. Waters. Realistic face modeling for animation. In *SIGGRAPH proceedings*, pages 55–62, 1995.
43. F. Ulgen. A step toward universal facial animation via volume morphing. In *6th IEEE International Workshop on Robot and Human communication*, pages 358–363, 1997.
44. B. Guenter, C. Grimm, D. Wood, H. Malvar, and F. Pighin. Making faces. *Proc.of ACM SIGGRAPH'98*, pages 55–66, 1998.
45. MJD Powell. Radial basis functions for multivariate interpolation: a review. *Algorithms for Approximation*, 1987.
46. T. Poggio and F. Girosi. A theory of networks for approximation and learning, 1989.
47. V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH proceedings*. ACM Press, 1999.

48. C.J. Kuo, R.S. Huang, and T.G. Lin. Synthesizing lateral face from frontal facial image using anthropometric estimation. In *Proceedings of International Conference on Image Processing*, volume 1, pages 133–136, 1997.
49. D. DeCarlo, D. Metaxas, and M. Stone. An anthropometric face model using variational technique. In *SIGGRAPH Proceedings*, 1998.
50. S. Gortler and M. Cohen. Hierarchical and variational geometric modeling with wavelets. In *Symposium on Interactive 3D Graphics*, pages 35–42, 1995.
51. W. Welch and A. Witkin. Variational surface modeling. In *SIGGRAPH Proceedings*, pages 157–166, 1992.
52. M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1987.
53. N.M. Thalmann, A. Cazedevals, and D. Thalmann. Modeling facial communication between an animator and a synthetic actor in real time. In *Proc. Modeling in Computer Graphics*, pages 387–396, 1993.
54. D. Terzopouos and R. Szeliski. Tracking with kalman snakes. *Active Vision*, pages 3–20, 1993.
55. K. Waters and D. Terzopoulos. Modeling and animating faces using scanned data. *Journal of Visualization and Computer Animation*, 2(4):123–128, 1990.
56. D. Terzopoulos and K. Waters. Techniques for realistic facial modeling and animation. In *Proc. of IEEE Computer Animation*, pages 59–74. Springer-Verlag, 1991.
57. I.S. Pandzic, P. Kalra, and N. M. Thalmann. Real time facial interaction. *Displays (Butterworth-Heinemann)*, 15(3), 1994.
58. E.M. Caldognetto, K. Vagges, N.A. Borghese, and G. Ferrigno. Automatic analysis of lips and jaw kinematics in vcv sequences. In *Proceedings of Eurospeech Conference*, volume 2, pages 453–456, 1989.
59. L. Williams. Performance-driven facial animation. In *Proc. of ACM SIGGRAPH '90*, pages 235–242. ACM Press, 1990.
60. E.C. Patterson, P.C. Litwinowicz, and N. Greene. Facial animation by spatial mapping. In *Proc. Computer Animation*, pages 31–44, 1991.
61. F. Kishino. Virtual space teleconferencing system - real time detection and reproduction of human images. In *Proc. Imagina*, pages 109–118, 1994.
62. P. Litwinowicz and L. Williams. Animating images with drawings. In *ACM SIGGRAPH Conference Proceedings*, pages 409–412, 1994.
63. L. Moubaraki, J. Ohya, and F. Kishino. Realistic 3d facial animation in virtual space teleconferencing. In *4th IEEE International workshop on Robot and Human Communication*, pages 253–258, 1995.
64. J. Ohya, Y. Kitamura, H. Takemura, H. Ishi, F. Kishino, and N. Terashima. Virtual space teleconferencing: Real-time reproduction of 3d human images. *Journal of Visual Communications an Image Representation*, 6(1):1–25, 1995.
65. BKP Horn and BG Schunck. Determining optical flow. *Artificial Intelligence*, pages 185–203, 1981.
66. T. Darrell and A. Pentland. Spacetime gestures. In *Computer Vision and Pattern Recognition*, 1993.
67. I.A. Essa, T. Darrell, and A. Pentland. Tracking facial motion, 1994.
68. J. Chai, J. Xiao, and J. Hodgins. Vision-based control of 3d facial animation. In *Proc. of Symposium on Computer Animation*, pages 193–206. ACM Press, 2003.

69. L. Zhang, N. Snavely, B. Curless, and S. M. Seitz. Spacetime faces: high resolution capture for modeling and animation. *ACM Trans. Graph.*, 23(3):548–558, 2004.
70. B. Choe B, H. Lee, and H.S. Ko. Performance driven muscle based facial animation. *The Journal of Visualization and Computer Animation*, 12(2):67–79, 2001.
71. J. Noh, D. Fidaleo, and U. Neumann. Gesture driven facial animation, 2002.
72. P. Joshi, W.C. Tien, M. Desbrun, and F. Pighin. Learning controls for blend shape based realistic facial animation. In *Eurographics/SIGGRAPH Symposium on Computer Animation*, pages 35–42, 2003.
73. Iso/iec 14496 - mpeg-4 international standard, moving picture experts group, www.cselt.it/mpeg.
74. J. Ostermann. Animation of synthetic faces in mpeg-4. In *Proc. of IEEE Computer Animation*, 1998.
75. M. Escher, I. S. Pandzic, and N. M. Thalmann. Facial deformations for mpeg-4. In *Proc. of Computer Animation'98*, pages 138–145, Philadelphia, USA, 1998. IEEE Computer Society.
76. S. Kshirsagar, S. Garchery, and N. M. Thalmann. Feature point based mesh deformation applied to mpeg-4 facial animation. In *Proc. Deform'2000, Workshop on Virtual Humans by IFIP Working Group 5.10*, pages 23–34, November 2000.
77. G. A. Abrantes and F. Pereira. Mpeg-4 facial animation technology: Survey, implementation, and results. *IEEE Transaction on Circuits and Systems for Video Technology*, 9(2):290–305, 1999.
78. F. Lavagetto and R. Pockaj. The facial animation engine: Toward a high-level interface for the design of mpeg-4 compliant animated faces. *IEEE Transaction on Circuits and Systems for Video Technology*, 9(2):277–289, 1999.
79. S. Garchery and N.M. Thalmann. Designing mpeg-4 facial animation tables for web applications. In *Proc. of Multimedia Modeling*, pages 39–59, 2001.
80. I. S. Pandzic. Facial animation framework for the web and mobile platforms. In *Proc. of the* 7^{th} *Int'l Conf. on 3D Web technology*, 2002.
81. I.S. Pandzic and R. Forchheimer. *MPEG-4 Facial Animation: The Standard, Implementation, and Applications*. John Wiley & Sons, 2002.
82. A. Pearce, B. Wyvill, G. Wyvill, and D. Hill. Speech and expression: A computer solution to face animation. In *Proc. of Graphics Interface'86*, pages 136–140, 1986.
83. J. P. Lewis. Automated lip-sync: Background and techniques. *Journal of Visualization and Computer Animation*, pages 118–122, 1991.
84. B. L. Goff and C. Benoit. A text-to-audovisual-speech synthesizer for french. In *Proc. of the Int'l. Conf. on Spoken Language Processing (ICSLP)*, pages 2163–2166, 1996.
85. P. Cosi, C. E Magno, G. Perlin, and C. Zmarich. Labial coarticulation modeling for realistic facial animation. In *Proc. of Int'l Conf. on Multimodal Interfaces 02*, pages 505–510, Pittsburgh, PA, 2002.
86. S. A. King and R. E. Parent. Creating speech-synchronized animation. *IEEE Trans. Vis. Graph.*, 11(3):341–352, 2005.
87. C. Pelachaud. Communication and coarticulation in facial animation. *Ph.D. Thesis, Univ. of Pennsylvania*, 1991.
88. J. Beskow. Rule-based visual speech synthesis. In *Proc. of Eurospeech 95*, Madrid, 1995.

89. E. Bevacqua and C. Pelachaud. Expressive audio-visual speech. *Journal of Visualization and Computer Animation*, 15(3-4):297–304, 2004.
90. Z. Deng, M. Bulut, U. Neumann, and S. S. Narayanan. Automatic dynamic expression synthesis for speech animation. In *Proc. of IEEE Computer Animation and Social Agents (CASA) 2004*, pages 267–274, Geneva, Switzerland, July 2004.
91. Z. Deng, J. P. Lewis, and U. Neumann. Synthesizing speech animation by learning compact speech co-articulation models. In *Proc. of Computer Graphics International*, pages 19–25, 2005.
92. Z. Deng, U. Neumann, J. P. Lewis, T. Y. Kim, M. Bulut, and S. Narayanan. Expressive facial animation synthesis by learning speech co-articulations and expression spaces. *IEEE Trans. Vis. Graph.*, 12(6):1523–1534, 2006.
93. C. Bregler, M. Covell, and M. Slaney. Video rewrite: Driving visual speech with audio. *Proc. of ACM SIGGRAPH'97*, pages 353–360, 1997.
94. E. Cosatto. Sample-based talking-head synthesis. *Ph.D. Thesis, Swiss Federal Institute of Technology*, 2002.
95. Y. Cao, P. Faloutsos, E. Kohler, and F. Pighin. Real-time speech motion synthesis from recorded motions. In *Proc. of Symposium on Computer Animation*, pages 345–353, 2004.
96. Y. Cao, P. Faloutsos, and F. Pighin. Expressive speech-driven facial animation. *ACM Trans. on Graph.*, 24(4), 2005.
97. E. Cosatto and H. P. Graf. Audio-visual unit selection for the synthesis of photo-realistic talking-heads. In *Proc. of ICME*, pages 619–622, 2000.
98. J. Ma, R. Cole, B. Pellom, W. Ward, and B. Wise. Accurate automatic visible speech synthesis of arbitrary 3d model based on concatenation of diviseme motion capture data. *Computer Animation and Virtual Worlds*, 15:1–17, 2004.
99. J. Ma, R. Cole, B. Pellom, W. Ward, and B. Wise. Accurate visible speech synthesis based on concatenating variable length motion capture data. *IEEE Transaction on Visualization and Computer Graphics*, 12(2):266–276, 2006.
100. Z. Deng and U. Neumann. efase: Expressive facial animation synthesis and editing with phoneme-level controls. In *Proc. of ACM SIGGGRAPH/Eurographics Symposium on Computer Animation*, pages 251–259, Vienna, Austria, 2006.
101. Z. Deng. Data-driven facial animation synthesis by learning from facial motion capture data. *Ph.D. Thesis, University of Southern California*, 2006.
102. S. Kshirsagar and N. M. Thalmann. Visyllable based speech animation. *Computer Graphics Forum*, 22(3), 2003.
103. E. Sifakis, A. Selle, A. R. Mosher, and R. Fedkiw. Simulating speech with a physics-based facial muscle model. In *Proc. of Symposium on Computer Animation (SCA)*, 2006.
104. M. Brand. Voice pupperty. *Proc. of ACM SIGGRAPH'99*, pages 21–28, 1999.
105. T. Ezzat, G. Geiger, and T. Poggio. Trainable videorealistic speech animation. *ACM Trans. Graph.*, pages 388–398, 2002.
106. V. Blanz, C. Basso, T. Poggio, and T. Vetter. Reanimating faces in images and video. *Computer Graphics Forum*, 22(3), 2003.
107. S. Kshirsagar, T. Molet, and N. M. Thalmann. Principal components of expressive speech animation. In *Proc. of Computer Graphics International*, 2001.
108. A. S. Meyer, S. Garchery, G. Sannier, and N. M. Thalmann. Synthetic faces: Analysis and applications. *International Journal of Imaging Systems and Technology*, 13(1):65–73, 2003.

109. Y. Cao, P. Faloutsos, and F. Pighin. Unsupervised learning for speech motion editing. In *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2003.
110. Q. Zhang, Z. Liu, B. Guo, and H. Shum. Geometry-driven photorealistic facial expression synthesis. In *Proc. of Symposium on Computer Animation*, pages 177–186, 2003.
111. D. Vlasic, M. Brand, H. Pfister, and J. Popović. Face transfer with multilinear models. *ACM Trans. Graph.*, 24(3):426–433, 2005.
112. E. Chang and O.C. Jenkins. Sketching articulation and pose for facial animation. In *Proc. of Symposium on Computer Animation (SCA)*, 2006.
113. J. Y. Noh and U. Neumann. Expression cloning. *Proc. of ACM SIGGRAPH'01*, pages 277–288, 2001.
114. R. W. Sumner and J. Popović. Deformation transfer for triangle meshes. *ACM Trans. Graph.*, 23(3):399–405, 2004.
115. H. Pyun, Y. Kim, W. Chae, H. W. Kang, and S. Y. Shin. An example-based approach for facial expression cloning. In *Proc. of Symposium on Computer Animation*, pages 167–176, 2003.
116. E. S. Chuang, H. Deshpande, and C. Bregler. Facial expression space learning. In *Proc. of Pacific Graphics'2002*, pages 68–76, 2002.
117. E. Chuang and C. Bregler. Moodswings: Expressive speech animation. *ACM Trans. on Graph.*, 24(2), 2005.
118. J. B. Tenenbaum and W. T. Freeman. Separating style and content with bilinear models. *Neural Computation*, 12(6):1247–1283, 2000.
119. V. Vinayagamoorthy, M. Gillies, A. Steed, E. Tanguy, X. Pan, C. Loscos, and M. Slater. Building expression into virtual characters. In *STAR report, Proc. of Eurographics 2006*, 2006.
120. S. C. Khullar and N. Badler. Where to look? automating visual attending behaviors of virtual human characters. In *Proc. of Third ACM Conf. on Autonomous Agents*, pages 16–23, 1999.
121. R. Vertegaal, G. V. Derveer, and H. Vons. Effects of gaze on multiparty mediated communication. In *Proc. of Graphics Interface'00*, pages 95–102, Montreal,Canada, 2000.
122. R. Vertegaal, R. Slagter, G. V. Derveer, and A. Nijholt. Eye gaze patterns in conversations: There is more to conversational agents than meets the eyes. In *Proc. of ACM CHI 2001 Conference on Human Factors in Computing Systems*, pages 301–308, 2001.
123. S. P. Lee, J. B. Badler, and N. Badler. Eyes alive. *ACM Trans. Graph.(Proc. of ACM SIGGRAPH'02)*, 21(3):637–644, 2002.
124. Z. Deng, J. P. Lewis, and U. Neumann. Practical eye movement model using texture synthesis. In *Proc. of ACM SIGGRAPH 2003 Sketches and Applications*, San Diego, 2003.
125. Z. Deng, J. P. Lewis, and U. Neumann. Automated eye motion synthesis using texture synthesis. *IEEE Computer Graphics and Applications*, pages 24–30, March/April 2005.
126. C. Busso, Z. Deng, U. Neumann, and S. Narayanan. Natural head motion synthesis driven by acoustic prosody features. *Computer Animation and Virtual Worlds*, 16(3-4):283–290, July 2005.
127. C. Busso, Z. Deng, M. Grimm, U. Neumann, and S. Narayanan. Rigid head motion in expressive speech animation: Analysis and synthesis. *IEEE Transaction on Audio, Speech and Language Processing*, March 2007.

128. C. Pelachaud, N. Badler, and M. Steedman. Generating facial expressions for speech. *Cognitive Science*, 20(1):1–46, 1994.
129. J. Cassell, C. Pelachaud, N. Badler, M. Steedman, B. Achorn, T. Becket, B. Douville, S. Prevost, and M. Stone. Animated conversation: Rule-based generation of facial expression, gesture and spoken intonation for multiple conversational agents. In *Proc. of ACM SIGGRAPH'94*, pages 413–420, 1994.
130. H. P. Graf, E. Cosatto, V. Strom, and F. J. Huang. Visual prosody: Facial movements accompanying speech. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition(FG'02)*, Washington, D.C., May,2002.
131. M. Costa, T. Chen, and F. Lavagetto. Visual prosody analysis for realistic motion synthesis of 3d head models. In *Proc. of Int'l Conf. on Augmented, Virtual Environments and Three-Dimensional Imaging*, Ornos, Mykonos, Greece, 2001.
132. E. Chuang and C. Bregler. Performance driven facial animation using blendshape interpolation. *CS-TR-2002-02, Department of Computer Science, Stanford University*, 2002.
133. Z. Deng, C. Busso, S. S. Narayanan, and U. Neumann. Audio-based head motion synthesis for avatar-based telepresence systems. In *Proc. of ACM SIGMM 2004 Workshop on Effective Telepresence (ETP 2004)*, pages 24–30, New York, NY, Oct. 2004.