Technical Report : CUCS-045-07

Speech Enabled Avatar from a Single Photograph

Dmitri Bitouk* Shree K. Nayar†

Department of Computer Science, Columbia University
November, 2007

(a) Input photograph (b) Synthesized images

Figure 1: A photograph of a person and images of an interactive avatar created from the photograph.

ABSTRACT

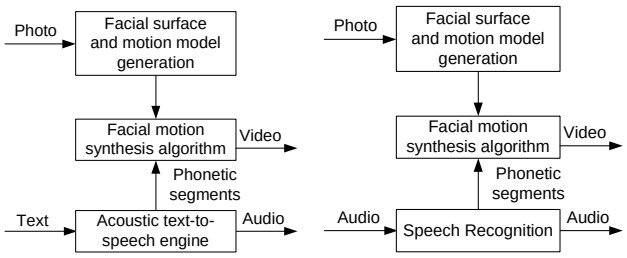
This paper presents a complete framework for creating speech-enabled 2D and 3D avatars from a single image of a person. Our approach uses a generic facial motion model which represents deformations of the prototype face during speech. We have developed an HMM-based facial animation algorithm which takes into account both lexical stress and coarticulation. This algorithm produces realistic animations of the prototype facial surface from either text or speech. The generic facial motion model is transformed to a novel face geometry using a set of corresponding points between the generic mesh and the novel face. In the case of a 2D avatar, a single photograph of the person is used as input. We manually select a small number of features on the photograph and these are used to deform the prototype surface. The deformed surface is then used to animate the photograph. In the case of a 3D avatar, we use a single stereo image of the person as input. The sparse geometry of the face is computed from this image and used to warp the prototype surface to obtain the complete 3D surface of the person's face. This surface is etched into a glass cube using sub-surface laser engraving (SSLE) technology. Synthesized facial animation videos are then projected onto the etched glass cube. Even though the etched surface is static, the projection of facial animation onto it results in a compelling experience for the viewer. We show several examples of 2D and 3D avatars that are driven by text and speech inputs.

Index Terms: H.5.2 [Information Interfaces and Presentation]: Multimedia Information Systems—Animations; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1 INTRODUCTION

While a substantial amount of work has been done on developing human face avatars, we have yet to see 2D and 3D avatars that are realistic in terms of animation as well as appearance. The goal of this paper is to create interactive 2D and 3D avatars of faces that provide realistic facial motion from text or speech inputs. Such speech-enabled avatars can significantly enhance user experience in a variety of applications including hand-held devices, information kiosks, advertising, news reporting and videoconferencing.

The major contribution of our work is an end-to-end system for building a 2D avatar from a photograph or a physical 3D avatar from a single stereo image of a face. Such avatars are animated from text or speech input with the help of a novel motion synthesis algorithm.



(a) Text-driven avatar animation (b) Speech-driven avatar animation

Figure 2: (a) Text- and (b) speech-driven visual speech synthesis. In both cases, the avatar is generated from a single photograph. The visual speech synthesis algorithm uses a face surface along with a facial motion model and a time-labeled phoneme segmentation derived from text or speech to produce a facial animation video. In the case of text-driven synthesis, phonetic segmentation is generated using acoustic text-to-speech synthesis. For speech-driven animation, we use speech recognition technology to obtain the spoken phonemes and their timings.

Our approach to facial animation employs the generic facial motion model previously introduced in [2]. The model represents a deformation of the 3D prototype facial surface due to articulation during speech as a linear combination of a small set of basis vector fields. The coefficients of this representation are the time-dependent facial motion parameters. In order to build a speaker-dependent facial motion model for a new subject, we firstly deform the prototype surface into a novel surface using a small set of feature points on the novel face. Then, the basis vector fields are adapted to the novel

*e-mail: bitouk@cs.columbia.edu
†e-mail:nayar@cs.columbia.edu

facial surface with the help of the deformation obtained in the previous step.

At each moment of time, the facial motion of a person is described by the small number of facial motion parameters described above. We train a set of Hidden Markov Models (HMMs) using the facial motion parameters obtained from motion capture data of a single speaker. Our facial motion synthesis algorithm utilizes the trained HHMs to generate facial motion parameters from either text or speech input, which are subsequently employed to produce realistic animations of avatars. Figure 2 shows the high-level architecture of our animation approach based on (a) text and (b) speech input.

We apply the facial motion synthesis algorithm to animate 2D avatars created from a single photograph. Since depth information is not directly available from a single photograph, we flatten (project) both the prototype surface and the basis vectors of the facial motion model to obtain a reduced 2D representation. To create a 2D avatar, we first select a few corresponding points between the prototype surface and the person's face in the photograph. Using these correspondences, we deform the prototype surface and adapt the basis vector fields to obtain the speaker-dependent facial motion model. We have developed real-time rendering software that can produce realistic facial animations of 2D avatars from facial motion parameters generated from either text or speech input. In order to enhance realism, our rendering system also synthesizes eye gaze motion and blinking. Figure 1 shows an input photograph and three sample frames of the speech-enabled avatar produced from it.

Another application considered in this paper deals with building volumetric displays featuring interactive 3D avatars. We present a simple method for recovering 3D face geometry and texture from a single catadioptric (mirror-based) stereo image [13]. Since a human face has large areas that are devoid of texture, stereo can only produce a sparse set of depth estimates. We present a fast method for obtaining the complete 3D surface of the face by deforming the prototype surface using the sparse depth estimates. A physical 3D avatar of the person's face is created by converting the obtained facial surface into a dense set of points, which are then engraved inside a solid glass block using sub-surface laser engraving (SSLE) technology [25]. The facial motion animation synthesized from text or speech is projected onto the static 3D avatar using a digital projector. Even though the physical shape of the avatar is static, the projection of facial animations onto it results in a compelling experience for the viewer.

2 Related Work

Our work is related to previous works in several fields, including, computer graphics, computer vision and 3D displays. Here we discuss the previous works that are most relevant to ours.

Facial Motion Representation: Existing approaches to facial motion synthesis fall into either image-based or model-based categories. Image-based approaches rely on building statistical models which relate temporal changes in the images at the pixel level to the text, or, equivalently, a sequence of phonemes uttered by the speaker. For instance, the Video Rewrite system [7] creates a database of phoneme-labeled mouth image patches. Given a novel audio track, the system selects and morphs images that match the spoken phonemes into a facial animation. MikeTalk [12] employs a low-dimensional representation of optical flow in order to blend between images corresponding to different phonemes. Image-based models cannot be employed for creating interactive avatars from a single photograph since they require a large training set of facial images in order to synthesize novel facial animations.

Model-based approaches typically represent the shape of a speaker's face with either a 2D or 3D mesh. Articulatory facial motion is described as deformation of the mesh and is controlled by a set of parameters. The deformed mesh, along with a texture image, is used to render facial animation. One of the most popular techniques parameterizes mesh deformations with the help of muscle models [35, 31, 17] which use facial muscle activations to produce facial animation. On the other hand, performance-driven approaches learn facial motion from recorded motions of people. Facial motion is usually recorded using optical motion capture [8] or structured light [24] techniques. In this paper, we take the model-based approach and use a compact parameterized facial model built from motion capture data presented in [2].

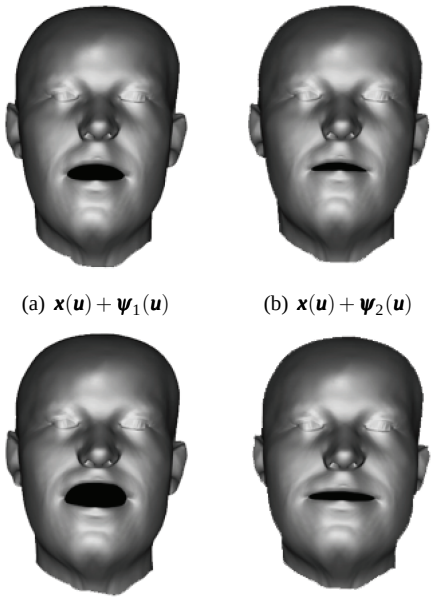
Facial Motion Synthesis: Given a parametric representation of facial motion, the role of speech synthesis algorithms is to generate parameter trajectories from a time-aligned sequence of phonemes. One of the approaches to visual speech synthesis is based on defining a key shape for each of the phonemes and smoothly interpolating between them [24]. The effects of coarticulation are taken into account using rule-based methods [10].

Similar to acoustic speech synthesis, visual speech synthesis methods fall either into concatenative or HMM-based categories. Concatinative approaches rely on stitching together pre-recorded motion sequences, which correspond to triphones [7], phonemes, syllables [19] or even longer speech units [8].

HMM-based synthesis [23], on the other hand, models the dynamics of visual speech with the help of hidden Markov models. Trajectories of facial motion parameters are generated from HMMs based on the maximum likelihood criteria. Brand [6], for example, builds a set of HMMs from audio and video data and employs HMM-based synthesis for speech-driven animation. Our method trains HMMs that can capture both the effects of coarticulation as well as lexical stress and produce realistic facial motions from either text or speech inputs.

Interactive 2D Avatars from a Photograph: Although a number of approaches to fitting a deformable model to a photograph have been suggested, generation of speech-enabled avatars from a single image remains an open research problem. For instance, Blantz et al [4] developed a method to transfer static facial expressions obtained from laser scans to photographs. The main drawback of this work is its high computational cost. A few commercial systems (see [1], for example) introduced recently aim to animate user-supplied facial images, but the facial motions they produce lack in realism. Our work builds an end-to-end system for creating interactive avatars from a single photograph which can be animated from text or speech in real-time. We believe the realism of the visual speech produced by our approach is fairly high compared to those of existing commercial systems.

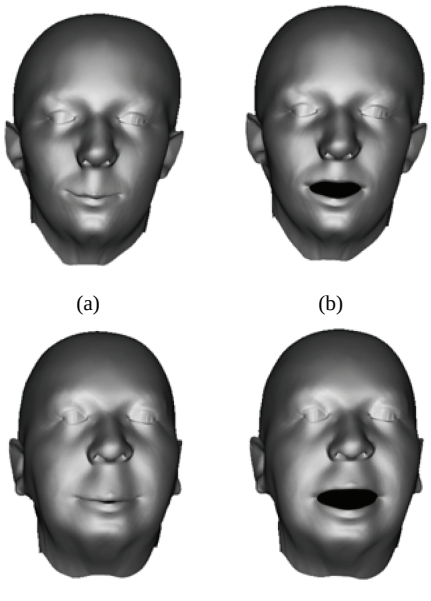
3D Face Reconstruction: Several approaches for recovering 3D geometry of human faces have been developed. Laser scanning [21] and structured light [37] techniques allow accurate recovery of facial shape [21], but require special hardware. Blantz and Vetter [3] proposed a morphable face model which can be fitted to face images. Photogrametric techniques [27] have been used to reconstruct face geometry from a set of manually marked points in multiple images. Stereo-based reconstruction of faces [22] is the closest one to the approach taken in this paper. We deform the prototype facial surface to match a sparse set of reconstructed 3D points. Our method for 3D geometry reconstruction provides an advantage since the estimated deformation of the prototype surface is also used to obtain a facial motion model that is adapted to the novel face, as described in Section 3.2.



(a) $\boldsymbol{x}(\boldsymbol{u})+\boldsymbol{\psi}_1(\boldsymbol{u})$ (b) $\boldsymbol{x}(\boldsymbol{u})+\boldsymbol{\psi}_2(\boldsymbol{u})$

(c) $\boldsymbol{x}(\boldsymbol{u})+\boldsymbol{\psi}_3(\boldsymbol{u})$ (d) $\boldsymbol{x}(\boldsymbol{u})+\boldsymbol{\psi}_4(\boldsymbol{u})$

Figure 3: Basis vector fields. (a)-(d) Action of the basis vector fields $\boldsymbol{\psi}_k(\boldsymbol{u}),\ k=1,2,...,4$ on the prototype facial surface.



(a) (b)

(c) (d)

Figure 4: Generic facial motion model shown in Figure 3 is adapted to two novel faces. Each of the columns (a)-(c) and (b)-(d) was generated using the same facial motion parameters. (a)-(b) Facial motions of the novel subject 1. (c)-(d) Facial motions of the novel subject 2.

3 Facial Motion Representation

Our approach to synthesizing facial animation from text or speech utilizes the 3D parametric facial motion model previously introduced in [2]. For the sake of completeness, this section briefly reviews this model, which is based on representing facial motions as deformations of a 3D surface which describes the geometry of a speaker's face. First, a generic, speaker-independent facial motion model is estimated. Then, the speaker-independent model is adapted to a novel speaker's face.

3.1 Generic Facial Motion Model

The generic face motion model describes deformations of the prototype face represented by a parametrized surface $\boldsymbol{x}(\boldsymbol{u}), \boldsymbol{x}\in\mathbb{R}^3, \boldsymbol{u}\in\mathbb{R}^2$. The displacement of the deformed face shape $\boldsymbol{x}_t(\boldsymbol{u})$ at the moment of time t during speech is represented as a linear combination of the basis vector fields $\boldsymbol{\psi}_k(\boldsymbol{u})$:

$$\boldsymbol{x}_t(\boldsymbol{u})=\boldsymbol{x}(\boldsymbol{u})+\sum_{k=1}^{N}\alpha_{k,t}\boldsymbol{\psi}_k(\boldsymbol{u}). \tag{1}$$

Vector fields $\boldsymbol{\psi}_k(\boldsymbol{u})$ defined on the prototype facial surface $\boldsymbol{x}(\boldsymbol{u})$ describe the principal modes of facial motion and are illustrated in Figure 3.1. The basis vector fields $\boldsymbol{\psi}_k(\boldsymbol{u})$ are learned from motion capture data as described in [2]. At each moment of time, the deformation of the prototype facial surface is completely described by a vector of facial motion parameters $\boldsymbol{\alpha}_t=\left(\alpha_{1,t},\alpha_{2,t},...,\alpha_{N,t}\right)^T$. The dimensionality of the facial motion model is chosen to be $N=9$.

3.2 Adapting Generic Facial Motion to a Novel Face

The above basis vector fields are defined with the respect to the prototype surface and, thus, have to be adjusted to match the geometry of a novel face. While this problem was addressed previously [26, 30], the approach described below enables one to map the generic facial motion model using a few corresponding points between the generic mesh and the novel face geometry.

We employed the method developed in [2] for facial motion transfer between different face geometries which is based on shape analysis using diffeomorphisms $\phi:\mathbb{R}^3\mapsto\mathbb{R}^3$ defined as continuous one-to-one mappings of $\mathbb{R}^3$ with continuously differentiable inverses. A diffeomorphism ϕ which transforms the source surface $\mathbf{x}^{(s)}(\mathbf{u})$ into the target surface $\mathbf{x}^{(t)}(\mathbf{u})$ can be computed from a sparse set of correspondences between the two surfaces [16].

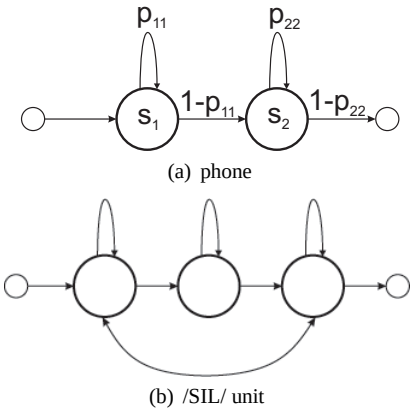
The diffeomorphism ϕ which carries the source surface into the target surface defines a non-rigid coordinate transformation of the embedding Euclidean space. Therefore, the action of the diffeomorphism ϕ on the basis vector fields $\boldsymbol{\psi}_k^{(s)}$ on the source surface is defined by the Jacobian of ϕ [5]:

$$\boldsymbol{\psi}_k^{(s)}(\mathbf{u})\mapsto D\phi\Big|_{\mathbf{x}^{(s)}(\mathbf{u}_i)}\cdot\boldsymbol{\psi}_k^{(s)}(\mathbf{u}), \tag{2}$$

where $D\phi\Big|_{\mathbf{x}^{(s)}(\mathbf{u}_i)}$ is the Jacobian of ϕ evaluated at the point $\mathbf{x}^{(s)}(\mathbf{u}_i)$:

$$(D\phi)_{ij}=\frac{\partial\phi_i}{\partial x_j},\ i,j=1,2,3. \tag{3}$$

Equation (3) is used to adapt the generic facial motion model to the geometry of a novel face. Given a set of corresponding feature points on the prototype and novel face shapes, we first estimate the diffeomorphism ϕ between them using the method presented in Appendix A. Then, the Jacobian $D\phi$ can be explicitly computed at any point on the generic face mesh and applied to the facial motion basis vector fields $\boldsymbol{\psi}_k^{(s)}$ in order to obtain the adapted basis vector fields. Figure 4 shows the results of applying this approach to two novel faces.



(a) phone

(b) /SIL/ unit

Figure 5: HMM topology for the visual speech. (a) HMM for a phone in the CMU set. The allowed transition between the HMM states s_1 and s_2 are shown as arcs with the transition probabilities p_{ij}. (b) The topology of the silence HMM /SIL/ (the state labels and transition probabilities are not displayed).

3.3 Visual Speech Unit Selection

In large vocabulary speech applications, uttered words are considered to be composed of phones which are acoustic realizations of phonemes. We make use of the CMU phone set, which consists of 39 distinct phones along with a non-speech unit /SIL/ which describes inter-word silence intervals. In order to accommodate lexical stress, the most common vowel phonemes are cloned into stressed and unstressed phones (for example, /AA0/ and /AA1/). In particular, we chose to model both stressed and unstressed variants of phones /AA/, /AE/, /AH/, /AO/, /AY/, /EH/, /ER/, /EY/, /IH/, /IY/, /OW/ and /UW/. The rest of the vowels in CMU set are modeled independent of their lexical stress.

Each of the phones, including stressed and unstressed variants, is represented as a 2-state HMM , while the /SIL/ unit is described using 3-state topology, as shown in Figure 3.3. The HMM states s_1 and s_2 explicitly represent the onset and end of the corresponding phone. The output probabilities of each HMM state is assumed to be given by a Gaussian distribution over the facial parameters $\boldsymbol{\alpha}_t$ which correspond to the HMM observations.

3.4 HMM Training

The goal of the HMM training procedure is to obtain maximum-likelihood estimates of the transition probabilities between HMM states and the sufficient statistics of the output probability densities for each HMM state from a set of observed facial motion parameter trajectories $\boldsymbol{\alpha}_t$ corresponding to the known sequence of words uttered by a speaker. As a training set, we utilize facial motion parameter trajectories derived from motion capture data obtained in [2].

In order to accommodate for the dynamic nature of visual speech, we augment the original facial motion parameters $\boldsymbol{\alpha}_t$ with their first and second derivatives. Our implementation of HMM training is based on the Baum-Welch algorithm [28] and similar in spirit to the embedded re-estimation procedure [36]. Overall, the HMM training is realized in three major steps.

Firstly, a set of monophone HMMs is trained. Secondly, in order to capture co-articulation effects, monophones models are cloned into triphone HMMs which explicitly take into account left and right neighboring phones. Finally, we employ decision-tree based clustering of triphone states to improve robustness of the estimated HMM parameters and predict triphones that were not seen in the training set.

S	IH0	K	Y	UW1	R	IH0	T	IY0
0	100	160	240	290	320	390	440	520

Table 1: Phone labels and their start times (in milliseconds) corresponding to the utterance “security.”

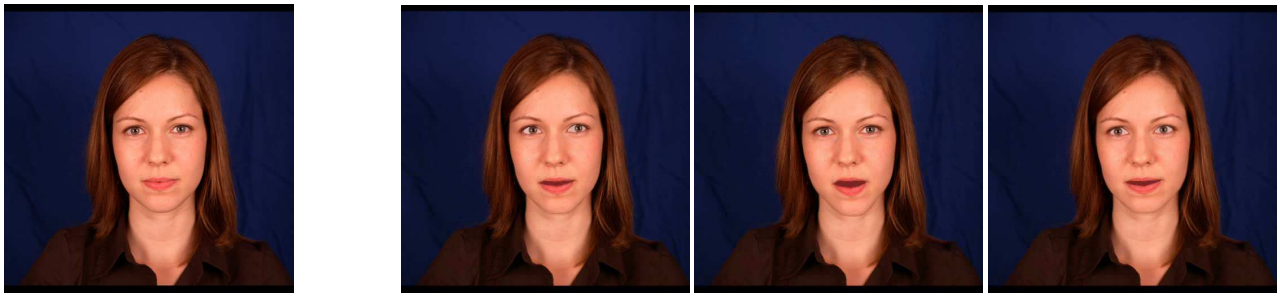
The training data consist of facial motion parameter trajectories $\boldsymbol{\alpha}_t$ and the corresponding word-level transcriptions. For the sake of convenience, the training set was manually segmented into a number of sentences. The dictionary employed in the HMM training process provides two instances of phone-level transcriptions for each of the words – the original transcription and a variant which end with the silence unit /SIL/. The output probability densities of monophone HMM states are initialized as a Gaussian density with mean and covariance equal to the global mean and covariance of the training data. Subsequently, 6 iterations of the Baum-Welch re-estimation algorithm are performed in order to refine the HMM parameter estimates using transcriptions which contain the silence unit only at the beginning and the end of each utterance. As the next step, we apply the forced alignment procedure [36] to obtain hypothesized pronunciations of each utterance in the training set. The final monophone HMMs are constructed by performing 2 iteration of the Baum-Welch algorithm.

In order to capture the dependence of a phone’s realization on the context and co-articulation, we utilize triphones, which take into account the proceeding and the following phones, as the speech units. The triphone HMMs are initialized by cloning the corresponding monophone models and are consequently refined by performing 2 iterations of Baum-Welch algorithm. The triphone state models are clustered with the help of tree-based procedure to reduce the dimensionality of the model and construct models for triphones unseen in the training set. The resulting models are often referred to as tied-state triphone HMMs in which the means and variances are constrained to be the same for triphone states belonging to a given cluster. The final set of tied-state triphone HMMs is obtained by applying another 2 iterations of the Baum-Welch algorithm.

4 Facial Motion Synthesis from Text and Speech

In order to synthesize trajectories of facial motion parameters $\boldsymbol{\alpha}_t$ either from text or acoustic speech signal, we firstly generate a sequence of time-labeled phones, as shown in Table 1. When text is used as input, we employ an acoustic text-to-speech (TTS) engine for the purpose of generating a waveform and the sequence of phones synthesized along with their corresponding start and end times. To synthesize facial animation from acoustic speech input, we utilize a speech recognizer [29] and use the forced alignment procedure [36] to obtain time-labels of the phones in the best hypothesis generated by the speech recognizer.

In the beginning of the synthesis stage, we convert the time-labeled phone sequence to an ordered set of context-dependent HMM states. Vowels are substituted with their lexical stress variants according to the most likely pronunciation chosen from the dictionary with the help of a monogram language model. In turn, we create an HMM chain for the whole utterance by concatenating clustered HMMs of each triphone state from the decision tree constructed during the training stage. The resulting sequence consists of triphones and their start and end times. Since each triphone unit is modeled as a two-state HMM, the start and end times of HMM states cannot be directly obtained from phone-level segmentation.



(a) Input image (b) Synthesized images

Figure 6: A photograph of a person and images of an interactive avatar created from it.

However, state-level segmentation can be inferred in the maximum-likelihood sense by utilizing the state transition probabilities estimated during HMM training stage.

The mean durations of the HMM states s_1 and s_2 with transition probabilities shown on Figure 3.3(a) can be computed as $p_{11}/(1-p_{11})$ and $p_{22}/(1-p_{22})$. If the duration of a triphone n described by a 2-state HMM in the phone-level segmentation is t_N, the durations $t_n^{(1)}$ and $t_n^{(2)}$ of its HMM states are proportional to their mean durations and are given by

$$t_n^{(1)} = \frac{p_{11} - p_{11}p_{22}}{p_{11} + p_{22} - p_{11}p_{22}} t_n, \quad t_n^{(2)} = \frac{p_{22} - p_{11}p_{22}}{p_{11} + p_{22} - p_{11}p_{22}} t_n. \quad (4)$$

Equation (4) for estimating HMM state durations allows us to convert a phone-level segmentation of the utterance into a time-labeled sequence of triphone HMM states $s^{(1)}, s^{(2)}, ..., s^{(N_S)}$. Smooth trajectories of facial motion parameters $\hat{\boldsymbol{\alpha}}_t = \left(\alpha^{(1)}, ..., \alpha^{(N_P)}\right)^T$ corresponding to the above sequence of HMM states is generated using the variational spline approach described in Appendix B.

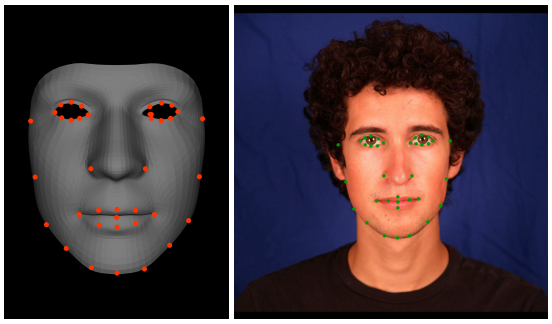
5 Interactive 2D Avatars

In this section we apply our facial motion synthesis algorithm to building a speech enabled avatar from a single photograph of a person. First, we deform the prototype surface to match the shape of a person's face in the photograph and adapt the facial motion using approach presented in Section 3.2. Next, to enhance realism, we use an algorithm that synthesizes eye gaze motion and eye blinking.

5.1 Fitting a Generic Facial Model to a Photograph

Since depth information cannot be recovered from a photograph, we use a reduced 2D representation. Both the prototype surface and basis vector fields of the generic facial motion model are flattened using orthogonal projection onto the canonical frontal view plane. In such reduced representation, avatars are 2D surfaces with facial motions which are restricted to the avatar's plane.

We start with a photograph of a person looking at the camera with a neutral facial expression. In order to establish correspondance between the generic facial model and subject's face, we manually mark a number of feature points on the photograph, as illustrated in Figure (7). Using the corresponding feature points, the generic mesh is deformed to fit the geometry of a subject's face in the photograph. The obtained deformation is used to transfer the generic motion model onto the deformed mesh using the approach presented in section 3.2.



(a) Prototype face surface (b) A photograph of a subject

Figure 7: A set of manually selected corresponding features on (a) the prototype face model and (b) the novel face photograph.

5.2 Eye Texture and Gaze Motion Synthesis

Changes in eye gaze direction can provide a compelling life-like appearance to avatars. Since some regions of the iris and the sclera are obstructed by the eyelids in the input photograph (see Figure 8 (a)), our approach automatically generates a new texture image for each eyeball. We use a sampling-based texture synthesis algorithm [11] to create the missing parts of the cornea and the sclera, as shown in Figure 8 (b). Using the points marked around the eyes, we first extract image regions which contain the eyeballs. Then, the position and shape of the iris are found using generalized Hough transform [18] in order to segment of the eye region into the iris and the sclera. Finally, a new eyeball image is generated by synthesizing missing texture inside the iris and sclera image regions.

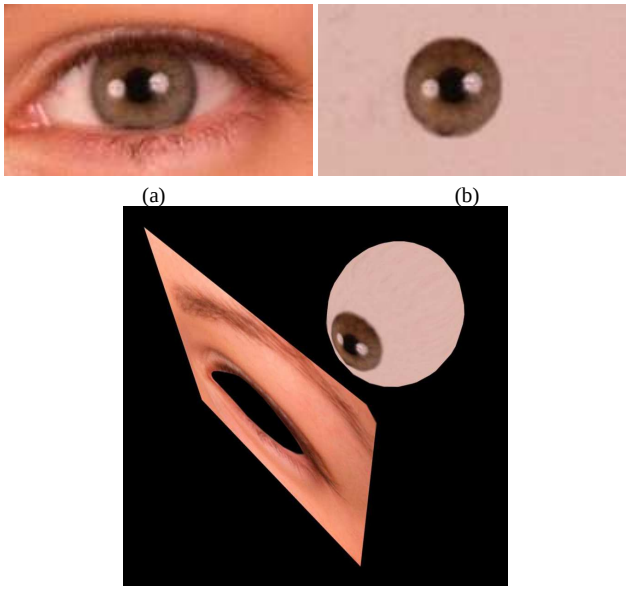
We model each eyeball as a textured sphere placed behind the eyeless face surface, as shown in Figure 8 (c). The eye gaze motion is generated by rotating the eyeballs around its centers. We use a previously proposed stochastic model [20] to generate the eye gaze changes.

5.3 Examples of 2D Avatars

We developed a real-time rendering software which creates realistic face animations of 2D avatars from text input. Figures 1 and 6 display a few sample frames from speech-enabled, 2D avatars synthesized using the approach presented above.

6 3D Avatars Using Sub-surface Laser Engraving

Facial motion representation and synthesis approaches presented in the previous section can be applied to building physical, 3D avatars.



(a) (b) (c)

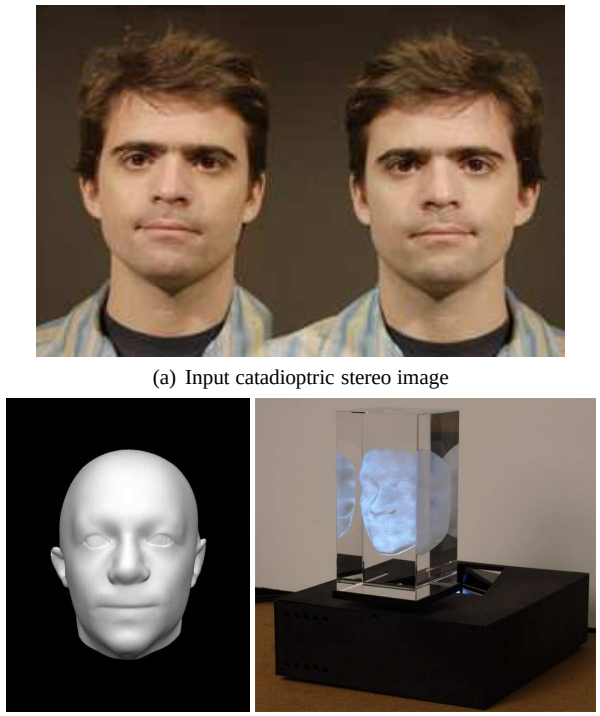
Figure 8: Eye texture synthesis and rendering. (a) A cropped image of the subject's eye. (b) New eyeball texture image that includes synthesized parts of the cornea and the sclera. (c) The eyeball is placed behind an eyeless image of the face and it is rotated to synthesize eye gaze changes.

Following the approach of Nayar and Anand [25], we build a 3D avatar of a specific person from a single stereo image using the technique of relief projection. We estimate the shape of a person's face and etch it inside a $100 \times 100 \times 200$ mm glass block. Facial animation, synthesized either from text or speech, is then projected onto the shape inside the glass block using a digital projector. Although the physical shape is static, the facial animations projected onto it result in a compelling experience for the viewer.

6.1 3D Face Model Acquisition Using Catadioptric Stereo

Using the method of [13], we capture a single image of a subject using a camera and a planar mirror. An example image is displayed on Figure 9 and includes the view of the subject as well as his/her reflection in the mirror. We treat the original and virtual (reflected) views as a stereo pair and subsequently rectify them in order to align the epipolar lines with the horizontal scan-lines. In order to calculate the camera's intrinsic parameters and the relative position of the mirror, the system is calibrated with help of Zhang's algorithm [38].

Due to the fact that images of a human face have large areas that are devoid of texture, our reconstruction algorithm recovers depth estimates for a sparse set of points. A dense mesh is generated by warping the prototype facial surface to match the sparse set of reconstructed points. First, we detect a number of Harris features [15] in both the direct and reflected views. Then, the detected features in each view are matched to locations in the second rectified view using normalized cross-correlation. We reconstruct depth of the image features from the obtained correspondences between the views using triangulation. Since correspondence between the reconstructed points and vertices on the generic face mesh is not given, we apply non-rigid iterative-closest point (ICP) algorithm similar to the one developed by [9] in order to warp the generic mesh. To initialize our ICP procedure, we manually mark a small



(a) Input catadioptric stereo image

(b) 3D face shape (c) Face shape etched inside a glass block

Figure 9: Volumetric 3D avatars created from a single stereo image. (a) Catadioptic stereo image of a person which include the direct view as well as a reflection in a planar mirror. (b The prototype facial surface deformed to match the geometry of the subject's face. (c) The face shape engraved inside a glass block on top of the projection system.

number of correspondences between points on the generic mesh and points in the observed views. These correspondences are then used to obtain an initial estimate of the rigid pose and warping of the generic mesh.

6.2 Interactive Volumetric Displays

We convert the estimated shape of a subject's face into a dense set point, namely a point cloud, which contains approximately 1.5 million points. The pointcloud is then engraved inside a solid piece of optical glass using SSLE technology [32]. Figure 9 (c) shows an example of a glass cube with the shape of a subject's face etched inside the cube. A facial animation video which is synthesized either from text or speech is relief-projected onto a static face shape inside the glass cube using a digital projection system. Although the shape is static, the projection of a face animation video onto it results in a realistic and compelling experience for an observer.

Due to differences in optical paths between the projector and individual scatters inside the glass, the projected frame has to be aligned and warped to match the engraved face shape. We use method developed by Nayar and Anand [25] to warp the projected video frames. Figure 6.1 shows sample views of interactive 3D avatars.

7 Discussion

In this paper, we developed an end-to-end system for creating speech-enabled avatars. Such avatars are built from a single photograph or, in the case of the volumetric displays, a single stereo image of the person. We also presented a method for generating

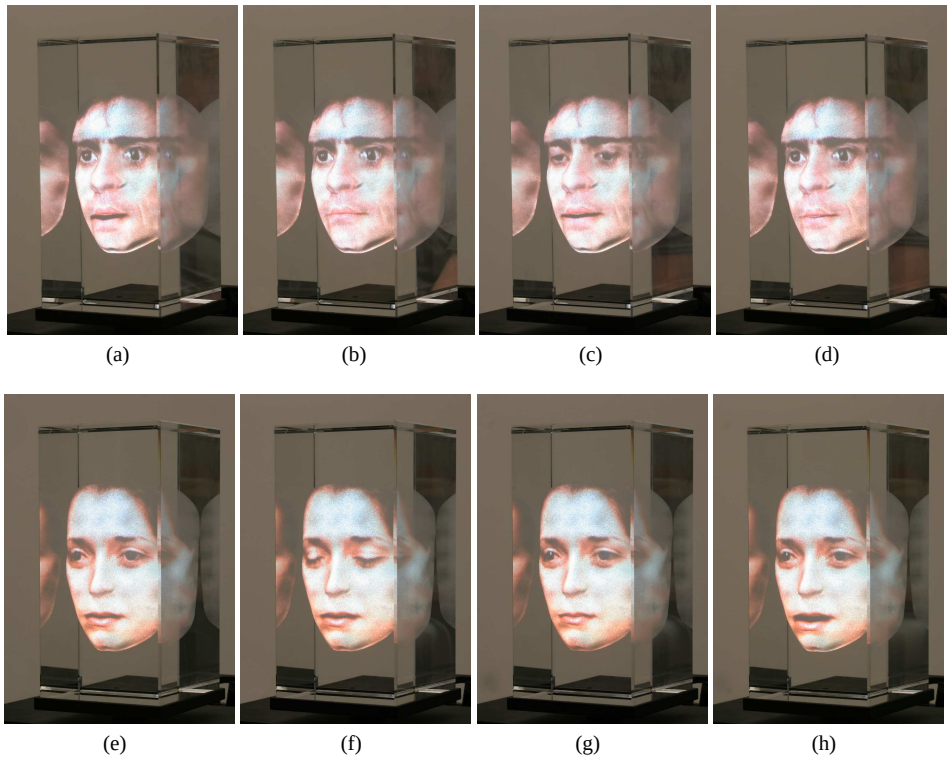


Figure 10: An example of synthesized facial animations projected onto the face shape etched inside a glass cube. Panels (a)-(d) and (e)-(h) show synthesized video frames for a male and a female subjects .

realistic facial motion from text and speech inputs. We now discuss the limitation of our work and open problems to be addressed in the future.

The HMM-based facial motion synthesis approach implicitly assumes that the visual and acoustic realizations of phonemes are synchronous. However, there exists cognitive evidence that there exist only loose synchronicity between them [14]. For example, facial articulations sometimes precede the sounds they produce. One may expect an improvement in the quality of synthesized facial animations if the visual and acoustic speech is modeled asynchronously by extending the HMM-based approach using, for example, dynamic Bayesian networks.

In order to transform the generic facial motion model and obtain the geometry of a novel face from a photograph, our approach requires a few corresponding points to be established between the generic surface and the novel face. In our current implementation, the corresponding features are marked manually on the input photograph. We expect that, in the future, 2D avatars can be created from photographs automatically using feature detection algorithms, such as [33].

Appearance of 2D avatars can be improved by synthesizing changes in the head pose. Although arbitrary rigid motions of the head cannot be created due to the lack of depth information, small head motions can be simulated using image warping methods. Perceived realism of our speech enabled avatars is also limited by the lack of facial emotions in the produced animations. Extending our approach to include synthesis of facial emotions is an open problem that we plan to address in the future work.

APPENDIX

A ADAPTATION OF THE GENERIC FACIAL MOTION MODEL TO A NOVEL FACE

In order to estimate a deformation between the prototype and the novel face surfaces, they are firstly aligned using rigid registration. Given a set of corresponding points $\mathbf{x}_1^{(s)}, \mathbf{x}_2^{(s)}, \ldots, \mathbf{x}_{N_p}^{(s)}$ on the prototype surface and $\mathbf{x}_1^{(t)}, \mathbf{x}_2^{(t)}, \ldots, \mathbf{x}_{N_p}^{(t)}$ on the aligned novel surface, the diffeomorphism between them is given by

$$\phi(\mathbf{x}) = \mathbf{x} + \sum_{k=1}^{N_p} K(\mathbf{x}, \mathbf{x}_k^{(s)}) \boldsymbol{\beta}_k \tag{5}$$

where the kernel $K(\mathbf{x}, \mathbf{y})$ was chosen to be as follows

$$K(\mathbf{x}, \mathbf{y}) \propto \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right) I_{3x3}. \tag{6}$$

and $\boldsymbol{\beta}_k \in \mathbb{R}^3$ are coefficients found by solving a system of linear equations [16].

In order to illustrate the transformation rule (3), let us consider the diffeomorphism ϕ which carries the source surface $\bar{\boldsymbol{x}}^{(s)}(\boldsymbol{u})$ into the target surface $\bar{\boldsymbol{x}}^{(t)}(\boldsymbol{u})$, $\phi\left(\mathbf{x}^{(s)}(\mathbf{u})\right) = \phi\left(\mathbf{x}^{(t)}(\mathbf{u})\right)$. The goal of the adaptation algorithm is to transfer the basis vector fields $\boldsymbol{\psi}_k^{(s)}(\mathbf{u})$ into the vector fields $\boldsymbol{\psi}_k^{(t)}(\mathbf{u})$ on the target surface such that the parameters $\boldsymbol{\alpha}_k$ of Equation (1) are invariant to difference in shape and proportions between the two surfaces which are described by

the diffeomorphism ϕ

$$\phi\left(\bar{\mathbf{x}}^{(s)}(\mathbf{u})+\sum_{k=1}^{N}\alpha_{k,t}\boldsymbol{\psi}_k^s(\boldsymbol{u})\right)=\bar{\mathbf{x}}^{(t)}(\mathbf{u})+\sum_{k=1}^{N}\alpha_{k,t}\boldsymbol{\psi}_k^t(\boldsymbol{u}). \quad (7)$$

Approximating the left-hand side of Eq. () using Taylor series up to the first order terms yields

$$\phi(\bar{\mathbf{x}}^{(s)}(\mathbf{u}))+\sum_{k=1}^{N}\alpha_{k,t}D\phi\Big|_{\mathbf{x}^{(s)}(\mathbf{u}_i)}\cdot\boldsymbol{\psi}_k^s(\boldsymbol{u})\approx\bar{\mathbf{x}}^{(t)}(\mathbf{u})+\sum_{k=1}^{N}\alpha_{k,t}\boldsymbol{\psi}_k^t(\boldsymbol{u}). \quad (8)$$

Since the above has to hold for all (small) values of $\boldsymbol{\alpha}_t$, the basis vector fields adapted to the target surface are given by

$$\boldsymbol{\psi}_k^{(t)}(\mathbf{u})=D\phi\Big|_{\mathbf{x}^{(s)}(\mathbf{u}_i)}\cdot\boldsymbol{\psi}_k^{(s)}(\mathbf{u}). \quad (9)$$

Jacobian $D\phi$ can be explicitely computed using Equation (5).

B Facial Motion Parameter Trajectory Smoothing

The problem of generating a trajectory of observation vectors which, in our case, correspond to the facial motion parameters $\boldsymbol{\alpha}_t$ from a given sequence of HMM states was considered in [6, 23] using the maximum-likelihood estimation. However, in our experience, this technique produces temporal discontinuities in the resulting trajectories of the facial motion parameters. The method presented below solves this problem by employing a variational approach with an explicit smoothness penalty.

Let N_F be the number of frames in the utterance, $t_1,t_2,...,t_{N_F}$ be the centers of each frames and $s_{t_1},s_{t_2},...,s_{t_{N_F}}$ be the sequence of HMM state corresponding to each frame. From the statistical point of view, the values of the facial motion parameters at the moments of time $t_1,t_2,...,t_{N_F}$ are characterized by the mean $\boldsymbol{\mu}_{t_1},\boldsymbol{\mu}_{t_2},...,\boldsymbol{\mu}_{t_{N_F}}$ and diagonal covariance matrices $\boldsymbol{\Sigma}_{t_1},\boldsymbol{\Sigma}_{t_2},...,\boldsymbol{\Sigma}_{t_{N_F}}$ of the corresponding HMM state output probability densities.

The vector components of a smooth trajectory $\hat{\boldsymbol{\alpha}}_t=\left(\alpha^{(1)},...,\alpha^{(N_P)}\right)^T$ of facial motion parameters is constructed as a solution to the following variational spline problem

$$\hat{\alpha}_t^{(k)}=\underset{\alpha_t^{(k)}}{\operatorname{argmin}}\sum_{n=1}^{N_F}\frac{\left(\alpha_{t_n}^{(k)}-\mu_{t_n}^{(k)}\right)^2}{\left(\sigma_{t_n}^{(k)}\right)^2}+\lambda\int_0^T\alpha_t^{(k)}\hat{L}\alpha_t^{(k)}dt \quad (10)$$

where $\mu_{t_n}^{(k)}$ are the components of $\boldsymbol{\mu}_{t_n}=\left(\mu_{t_n}^{(1)},\mu_{t_n}^{(1)},...,\mu_{t_n}^{(N_P)}\right)^T$, $\left(\sigma_{t_n}^{(k)}\right)^2$ are the diagonal components of $\boldsymbol{\Sigma}_{t_n}=diag\left(\left(\sigma_{t_1}^{(k)}\right)^2,\left(\sigma_{t_n}^{(2)}\right)^2,...,\left(\sigma_{t_n}^{(N_P)}\right)^2\right)$, $\hat{L}$ is a self-adjoint differential operator, and λ is the parameter controlling smoothness of the solution.

It follows from Wahba's Representer Theorem [34] that the solution to the variational problem (10) is given in the following form

$$\hat{\alpha}_t^{(k)}=\sum_{l=1}^{N_F}K(t_l,t)\beta_l, \quad (11)$$

where kernel $K(t_1,t_2)$ is the Green's function of the self-adjoint differential operator L. In our implementation, we chose kernel $K(t_1,t_2)$ to be Gaussian

$$K(t_1,t_2)\propto e^{-\frac{(t_2-t_1)^2}{2\sigma_K^2}}. \quad (12)$$

The vector of unknown coefficients $\boldsymbol{\beta}=(\beta_1,\beta_2,...,\beta_{N_F})^T$ that minimizes the right-hand side of equation (10) after the substitution (11) is the solution to the following system of linear equations

$$\left(\boldsymbol{K}+\lambda\boldsymbol{S}^{-1}\right)\boldsymbol{\beta}=\boldsymbol{\mu}, \quad (13)$$

where $\boldsymbol{K}$ is a $N_F\times N_F$ matrix with the elements $[\boldsymbol{K}]_{l,m}=K(t_l,t_m)$, $\boldsymbol{S}$ is a $N_F\times N_F$ diagonal matrix $\boldsymbol{S}=diag\left(\left(\sigma_{t_1}^{(n)}\right)^2,\left(\sigma_{t_2}^{(n)}\right)^2,...,\left(\sigma_{t_{N_F}}^{(n)}\right)^2\right)$ and $\boldsymbol{\mu}=\left(\mu_{t_1}^{(n)},\mu_{t_2}^{(n)},...,\mu_{t_{N_F}}^{(n)}\right)^T$.

References

[1] Crazy talk. www.reallusion.com/crazytalk/.

[2] D. Bitouk. *Head-pose and Illumination Invariant 3-D Audio-Visual Speech Recognition*. PhD thesis, The Johns Hopkins University, 2006.

[3] V. Blantz and T. Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, 1999.

[4] V. Blanz, C. Basso, T. Poggio, and T. Vetter. Reanimating faces in images and video. *Comput. Graph. Forum*, 22 (3):641–650, 2003.

[5] W. M. Boothby. *An introduction to differentiable manifolds and Riemannian geometry*. Academic Press, 1986.

[6] M. Brand. Voice puppetry. In *SIGGRAPH*, 1999.

[7] C. Bregler, M. Covell, and M. Slaney. Video rewrite: Driving visual speech with audio. In *SIGGRAPH 97, Computer Graphics Proceedings, Annual Conference Series*, 1997.

[8] Y. Cao, P. Faloutsos, E. Kohler, and F. Pighin. Real-time speech motion synthesis from recorded motions. In *Eurographics Symposium on Computer Animation*, 2004.

[9] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Comput. Vis. Image Underst.*, 89:114–141, 2003.

[10] M. Cohen and D. Massaro. Modeling coarticulation in synthetic visual speech. In *Models and Techniques in Computer Animation*. 1993.

[11] A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *IEEE International Conference on Computer Vision*. 1999.

[12] T. Ezzat, G. Geiger, and T. Poggio. Trainable videorealistic speech animation. In *SIGGRAPH 02, Computer Graphics Proceedings, Annual Conference Series*, 2002.

[13] J. Gluckman and S. K. Nayar. Catadioptric stereo using planar mirrors. *Int. J. Comput. Vision*, 44:65–79, 2001.

[14] K. Grant, V. van Wassenhoveb, and D. Poeppelb. Detection of auditory (cross–spectral) and auditory-visual (cross–modal) synchrony. *Speech Communication*, 44:43–54, 2004.

[15] C. Harris and M. Stephen. A combined corner and edge detector. In *Fourth Alvey Vision Conference*, 1988.

[16] S. Joshi and M. I. Miller. Landmark matching via large deformation diffeomorphisms. *IEEE Trans. on Image Processing*, 9:1357–1370, 2000.

[17] K. Kaehler, J. Haber, H. Yamauch, and H. Seidel. Head shop: Generating animated head models with anatomical structure. In *SIGGRAPH 02, Symposium on Computer Animation*, 2002.

[18] C. Kimme, D. Ballard, and J. Slansky. Finding circles by an array of accumulators. *Communications of the ACM*, 18:351–363, 1975.

[19] S. Kshirsagar and N. Thalmann. Visyllable based speech animation. In *Computer Graphics Forum (Eurographics Conference)*, 2003.

[20] S. P. Lee, J. P. Badler, and N. I. Badler. Eyes alive. In *International Conference on Computer Graphics and Interactive Techniques*, 2002.

[21] Y. Lee, D. Terzopoulos, and K. Walters. Realistic modeling for facial animation. In *SIGGRAPH*, 1995.

[22] R. Lengagne, P. Fua, and O. Monga. Using differential constraints to generate a 3d face model from stereo. In *14th International Conference on Pattern Recognition*, 1998.

[23] T. Masuko, T. Kobayashi, M. Tamura, J. Masubuchi, and K. Tokuda. Text-to-visual speech synthesis based on parameter generation from hmm. In *IEEE International Conference of Acoustics, Speech and Signal Processing*, 1998.

[24] P. Muller, G. Kalberer, M. Proesmans, and L. V. Gool. Realistic speech animation based on observed 3D face dynamics. *Vision, Image & Signal Processing*, 4, 2005.

[25] S. Nayar and V. Anand. 3d display using passive optical scatterers. *IEEE Computer*, July:54–63, 2007.

[26] J. Y. Noh and U. Neumann. Expression cloning. In *ACM SIGGRAPH*, 2001.

[27] F. Pighin, R. Szeliski, and D. H. Salesin. Modeling and animating realistic faces from images. *Int. J. of Computer Vision*, 50:143–169, 2002.

[28] L. Rabiner. *Readings in speech recognition*, chapter A tutorial on HMM and selected applications in speech recognition, pages 267–296. Morgan Kaufmann, 1990.

[29] K. Seymore, S. Chen, S. Doh, M. Eskenazi, E. Gouvea, B. Raj, M. Ravishankar, R. Rosenfeld, M. Siegler, R. Stern, and E. Thayer. CMU Sphinx-3 english broadcast news transcription system. In *DARPA Speech Recognition Workshop*, 1998.

[30] R. W. Summer and J. Popovic. Deformation transfer for triangle meshes. In *ACM SIGGRAPH*, 2004.

[31] D. Terzopoulos and K. Waters. Analysis and synthesis of facial image sequences using physical and anatomical models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15 (6), 1993.

[32] I. Troitski. *Stereoscopic Displays and Virtual Reality Systems XII*, chapter Laser-induced image technology (yesterday, today, and tomorrow), pages 293–301. SPIE, 2005.

[33] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[34] G. Wahba. *Spline Models for Observational Data*. Regional Conference Series in Applied Mathematics, SIAM, 1990.

[35] K. Waters. A muscle model for animating three-dimensional facial expressions. In *Computer Graphics*, 1987.

[36] S. Young, J. J, J. Odell, D. Ollason, and P. Woodland. *The HTK Book*. Cambridge University Press, 2005.

[37] L. Zhang, N. Snavely, B. Curless, and S. Seitz. Spacetime faces: High-resolution capture for modeling and animation. In *SIGGRAPH*, 2004.

[38] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1330–1334, 2000.