VIE 2003 SPECIAL SECTION

Speech-driven facial animation using a hierarchical model

D.P. Cosker, A.D. Marshall, P.L. Rosin and Y.A. Hicks

Abstract: A system capable of producing near video-realistic animation of a speaker given only speech inputs is presented. The audio input is a continuous speech signal, requires no phonetic labelling and is speaker-independent. The system requires only a short video training corpus of a subject speaking a list of viseme-targeted words in order to achieve convincing realistic facial synthesis. The system learns the natural mouth and face dynamics of a speaker to allow new facial poses, unseen in the training video, to be synthesised. To achieve this the authors have developed a novel approach which utilises a hierarchical and nonlinear principal components analysis (PCA) model which couples speech and appearance. Animation of different facial areas, defined by the hierarchy, is performed separately and merged in post-processing using an algorithm which combines texture and shape PCA data. It is shown that the model is capable of synthesising videos of a speaker using new audio segments from both previously heard and unheard speakers.

1 Introduction

Since the pioneering work of Parke [1] in the 1970s, the development of realistic computer-generated facial animation has received a vast amount of attention, crossing over into fields such as movies, video games, mobile and video communications and psychology. The problem not only encompasses the design of a mechanism capable of representing a face realistically, but also its control. Most computer-generated facial systems are based on 3-D models [2, 3] or image-based models [4, 5], and are parametric. Using these representations we may animate a face using only a speech signal. This is desirable for many applications, such as low-bandwidth network communications and broadcasts, movie lip resynching and lip synching for animated movies.

Traditional methods for producing automatic facial animation from speech include the input of phonemes (derived from text or speech) [2–7] or spectral speech features [8] into a model which in turn produces parameters used to drive the animation of the facial model. In previous work facial parameters have been obtained from suitable inputs using several techniques, including phoneme-to-viseme based mappings [2, 5, 6], rule-based methods [9] and nonlinear models (such as hidden Markov models (HMM) or neural networks) [8, 10, 11].

Perhaps the most popular means of automatically animating faces from speech is using models based on phoneme-to-viseme mappings, where the model controls the animation of the mouth, and the motion of the mouth during animation influences surrounding vertices representing the lower part of the face. Phonemes are discrete units of speech sound born from the study of phonetics. In modern literature, the visual counterpart of the phoneme has come to be known as the viseme. The desirable property of performing animation using phonemes/visemes is that once the phonetic labelling has been performed the audio-to-visual mapping becomes more constrained, i.e. given a phoneme we are then confident that an appropriate corresponding output mouth would belong to a known class of viseme. The performance of the resulting facial animation is then dependent on two remaining problems. These problems may be defined as follows:

(i) *Sophistication of the facial model being utilised.* This is perhaps one of the most important issues to address if the aim is to produce video or near-video realistic facial animation irrespective of the facial display method (image-based or 3-D based). The characteristics of a realistic 3-D facial model will invariably be a high polygon count with a sufficiently high number of control points (or feature points) in the model as to provide intricate and lifelike facial motions, and a realistic texture map [12]. A successfully realistic image-based model, however, relies on high-quality training images which are adequately normalised with respect to each other. The latter issue is perhaps more important than the former since without good image normalisation the visual output shows artefacts, such as jerkiness and texture flicker. While we are used to low-quality video in our everyday lives (e.g. low resolution/blocky) this differs from poor normalisation, which provides a psychological cue for us to categorise an animation immediately as being ‘unrealistic’.
(ii) *Phonetic-to-visual mapping.* While the first problem addressed is that of building a realistic output tool, the second problem is then of choosing visemes to correspond with phonemes and defining a scheme where coarticulation is addressed.

By examining the success of phonetic input automatic facial animation systems a question arises: ‘Is this then, in fact, the way to proceed in automatically animating faces from speech?’. The alternative method to phonetic inputs is to use

IEE Proceedings online no. 20040752

doi: 10.1049/ip-vis:20040752

Paper first received 30th October 2003 and in revised form 5th May 2004. Originally published online 29th July 2004

The authors are with the School of Computer Science, Cardiff University, Cardiff, CF24 3AX, UK

continuous speech signals. To date there is yet to be published a system using continuous speech which provides true video-realistic facial animation. It could be argued that this is what phonetic systems use; they simply process the continuous signal first to produce robust features. However, used on their own phonetic inputs are only capable of providing mouth animation and the problem of animation for the rest of the face is typically addressed *ad hoc* by stitching synthesised video onto real background footage [4, 6]. There is also the school of thought which observes that phonemes were introduced to help distinguish pronunciation and are not entirely suited for animation purposes, in which case there might exist a better method still with the phoneme replaced with a new unit optimally chosen for the purpose of mouth animation [8].

This makes continuous methods more appealing since it points to a more back-to-basics approach where information within the speech can be examined more closely. Research using continuous signals has also yielded several (often unexpected) advantages over phonetically driven systems, namely the animation of parts of the face other than the mouth associated with expression, sometimes correlated with perceived emotion in the speech. This leads to the intriguing proposition that by analysing continuous speech we may be able to not only animate the mouth realistically, but the whole face, directly and automatically from the continuous speech signal.

In this paper we present a system for producing facial animation based on a hierarchical facial model. To build the model we analyse speech as a continuous signal without making assumptions about phonetic content. The effect of this is that, as well as providing mouth animation, the model can also produce facial animations which exhibit expression. The system learns the facial dynamics of a speaker during a training phase and uses this as a foundation to synthesise novel facial animations. For training, a small corpus of audio and video is collected of a speaker uttering a list of words that target visually distinct mouth postures. After training, new speech can be supplied, by the original speaker or a new speaker, and synchronised video-realistic facial animation is produced. The final video is of the person used in the training phase.

To model mappings between input speech and output parameters we introduce a hierarchical nonlinear speech-appearance model built from data extracted from the training set. We decompose the face hierarchically where the root corresponds to a nonlinear model of the whole face and sub-nodes nonlinearly model smaller, more specific facial areas. Modelling hierarchically allows us to concentrate on learning the variation exhibited in sub-facial areas during speech independently from any variation in the rest of the face. The alternative, to model the face as a whole, would require an unrealistically large training set which contains every combination of facial movement due to emotion, head pose and speech. This approach has also been taken by Cosatto and Graf [13]. Modelling in this way also improves the specificity of statistical models, such as point distribution models (PDM) (which we employ), as demonstrated in [14] with a tracking algorithm. Figure 1 gives an example of a hierarchical facial model.

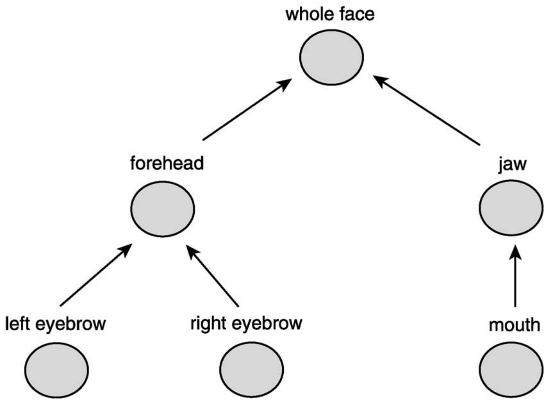

Fig. 1 *Hierarchical facial model overview*

2 Overview

Our system has four main stages: training, model learning, facial synthesis and video production. In the training phase a 25 frames/s video is captured of a speaker uttering a list of words targeting different visemes. Images extracted from the video are then annotated by placing landmarks at the boundaries of facial features. This may be done manually or semi-automatically [15–17]. The system then extracts the landmarks from the training set and builds a hierarchical model of the face. For the purpose of this article we only build a hierarchy which includes the face (as the root node) and the mouth. This reduced hierarchy is intended to primarily demonstrate lip-synch (performed by the mouth node).

Given our training set, we next extract the data required to build each node in the hierarchy. For the representation of a node we introduce a nonlinear speech-appearance model. This is an extension of an appearance model introduced by Cootes *et al.* [16] encoding relations between appearance parameters and speech vectors allowing the synthesis of facial configurations given new audio. The root node of the model is built using the full facial landmark and image data. For nodes such as the mouth we simply extract landmarks and texture associated with the area of interest. For representation of speech signals we process our training audio using mel-cepstral analysis [18] with 40 ms hamming windows, yielding 12 mel-cepstral coefficients per video training frame. To reduce the dimensionality of the speech we perform principal component analysis (PCA) on the feature set.

To achieve facial synthesis given new audio we first process the incoming audio using mel-cepstral analysis. After dimensional reduction of the speech the nonlinear speech-appearance model at each hierarchy node is used to synthesise a facial area. In the final stage, synthesised facial information from sub-nodes is then combined to construct an entire face. This is achieved using a novel algorithm for adding together shape and texture parameters in the image domain. The algorithm uses information in a node's parent to synthesise its own appearance in a parents-corresponding facial area. For example, a mouth parameter may use a facial model to produce a facial image which contains the appearance of the desired mouth. This is repeated until the root of the hierarchy is reached, which results in the synthesis of a full facial output frame.

3 Data acquisition

The training process requires a video of a speaker uttering a set of viseme-rich phrases with which to build our hierarchical model. We capture audio at 33 kHz mono using the on-camera microphone and video at 25 frames/s. Each image in our training set is then transformed into the YIQ colour space before being labelled with 82 landmarks between the top of the eyebrows and the jaw. We use the YIQ colour space since it separates grey-level information

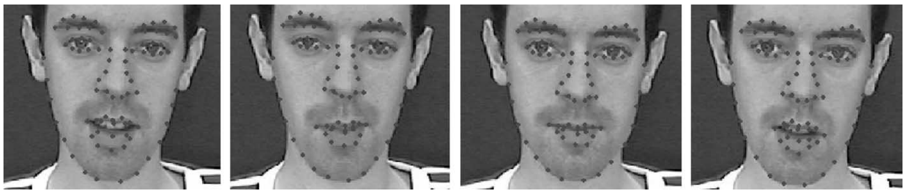

Fig. 2 *Example annotated training images*

(luminance) from colour information (hue and saturation) allowing us to build our synthesis model in grey-level and add the two colour channels during output. This essentially reduces the complexity of our synthesis model. The alternative would be to build a synthesis node using concatenated image colour channel information. Given the three colour channels for each image we store luminance vectors with the intention of building our synthesis model and hue and saturation vectors for adding colour later on. Figure 2 gives examples of labelled training images.

4 Hierarchical modelling

Facial area synthesis for each node in our hierarchical model is based on an appearance model [16]. The appearance model has become the most widely used image-based model in computer vision research today. Originally presented as a tool for image segmentation and classification, it also provides an ideal output for a facial animation system. This is due to its parametric nature where variation of elements in the appearance parameter relate to different facial variations. When attempting to animate the whole face by varying elements of the appearance parameter we initially see a drawback. We notice that variations are not isolated, i.e. variation in the mouth is accompanied by variation in another part of the face, e.g. the eyebrows or the eyes. The advantage of modelling the face hierarchically using an appearance model is that variation in different facial parts is analysed and modelled separately. We are far more likely to generalise and specify the training data better than if we were to use a single flat model of the face.

In order to avoid variations in different parts of the face interfering with the modelling process, the data collection phase used for building facial animation systems employing *flat* facial models requires a subject to maintain a neutral expression [4, 5]. This ensures that mouth animations are not accompanied by idiosyncratic behaviour in other parts of the face which may reappear during synthesis. For mouth animation alone, this is a sensible approach. However, it restricts animation to this end alone and does not allow modelling of facial expression, unless an unrealistically large training set is obtained where every combination of facial action is accounted for [13]. This is a second advantage to modelling hierarchically since it allows mixtures of variations in different facial areas to be present in the training set without interfering with the modelling stage.

Modelling hierarchically using eigenspaces has also been applied to other applications such as tracking [14] with similar benefits to those described above. The concept of a hierarchical model of the face was also hinted at in [8] as a means of providing a greater degree of freedom for animation.

4.1 *Initial facial area modelling*

Using data gathered during the training phase, we begin building our model by extracting landmark shape data, and shape-free texture data, for each facial area. The process for achieving this follows a similar approach to that described by Cootes *et al.* [16]. To build a synthesis node, we take landmark data belonging to the appropriate facial area and perform Procrustes alignment. When performing alignment we only align with respect to translation and rotation since we regard scale as extra shape information. We are able to do this since our training subject maintains a constant distance from the camera when recorded, and minimises out-of-plane head movements.

Taking the mean of the aligned shape data we warp each luminance colour channel image in the training set to this mean from the landmarks in the image plane to yield a training set of shape-free luminance patches.

Using this luminance data, and the captured audio data, we can then proceed to build a nonlinear speech-appearance model for that node. We first build an appearance model for each node defined using

$$\boldsymbol{x} = \bar{\boldsymbol{x}} + \boldsymbol{P}_s \boldsymbol{W}_s^{-1} \boldsymbol{Q}_s \boldsymbol{c} \tag{1}$$

$$\boldsymbol{g} = \bar{\boldsymbol{g}} + \boldsymbol{P}_g \boldsymbol{Q}_g \boldsymbol{c} \tag{2}$$

where $\boldsymbol{x}$ and $\boldsymbol{g}$ are examples of shape and texture, $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{g}}$ are the mean normalised shape and texture vectors, $\boldsymbol{P}_s$ and $\boldsymbol{P}_g$ are the eigenvectors of each training sample distribution, $\boldsymbol{b}_s$ and $\boldsymbol{b}_g$ are shape and texture parameters, $\boldsymbol{W}_s$ is the scale matrix and $\boldsymbol{Q}_s$ and $\boldsymbol{Q}_g$ are the shape and texture parts of the eigenvectors $\boldsymbol{Q}$. Using this model we then project each shape and texture vector associated with a node into appearance parameter space using

$$\boldsymbol{c} = \boldsymbol{Q}\boldsymbol{b} \tag{3}$$

giving us n appearance parameters $\boldsymbol{c}$ for a given node. More details are available in [16]. An example of the distribution of the two highest modes of appearance variation for a mouth node training set is shown in Fig. 3. Representation of these parameters in the image domain at ±2 standard deviations from the mean is shown in Fig. 4.

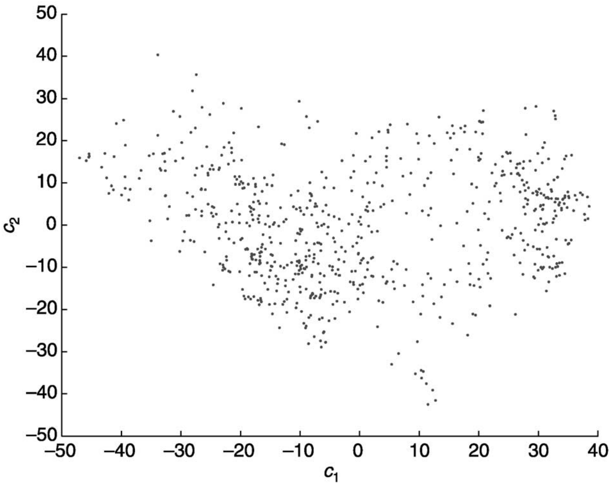

Fig. 3 *Distribution of mouth appearance parameters represented by the two highest modes of variation*

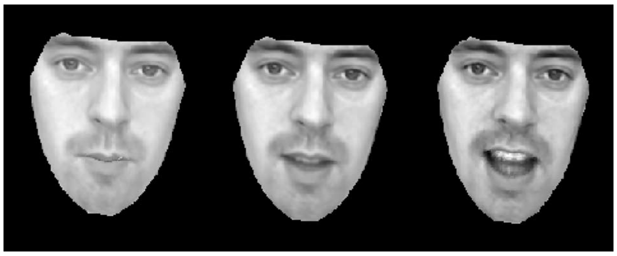

Fig. 4 *Highest mode of face appearance variation at + (left) and − (right) 2 standard deviations from the mean*

4.2 Nonlinear appearance modelling

By examining Fig. 3 we see that our appearance parameter distribution is nonlinear. It is well documented that fitting single linear models to nonlinear distributions reduces specificity and generalisation [19, 20]. We therefore model appearance parameter distributions relating to sub-facial areas with local-linear models. We perform k-means clustering on the appearance distribution yielding k appearance parameter clusters. In the following Section we describe how appearance parameters are related to speech parameters.

A note on our k-means clustering approach; in order to decide k we performed a series of experiments with varying values for k and randomly chosen initial centre positions. The final value for k and the starting positions for the k-means search were then chosen to be those yielding the lowest error value. We found that for best performance a relatively high number of centres is desirable; in our case we used no fewer than 60. However, this number is largely dependent on the size of the training set; a too small training set (less than 400 appearance parameters in our case) yields an unstable synthesis model which is highly sensitive to changes in k and starting centres. Generally, with enough data, we found that the importance of starting centre positions becomes irrelevant (we discovered this phenomena with distributions of over 700 appearance parameters).

4.3 Cluster modelling for associating appearance with speech

To reiterate our overall aim so far, we wish to encode relationships between our appearance model and our speech training set so that given speech we may estimate an appearance parameter $\boldsymbol{c}$ for a node's facial area. After clustering our appearance parameters we find that each cluster represents a specific kind of sub-facial variation. The problem now is to relate the sub-facial variation of each cluster with corresponding variation in speech.

To achieve this mapping we first reduce the dimensionality of our speech training set using principal component analysis yielding the model

$$\boldsymbol{a} = \bar{\boldsymbol{a}} + \boldsymbol{P}_a \boldsymbol{s} \tag{4}$$

where $\boldsymbol{a}$ is a speech vector, $\bar{\boldsymbol{a}}$ is the mean speech vector in our training set, $\boldsymbol{P}_a$ are the eigenvectors of our speech distribution and $\boldsymbol{s}$ is a speech parameter. Recall that our initial speech training set consists of a set of mel-cepstral coefficients calculated with a 40 ms window over the training speech, each vector of coefficients corresponding to an appearance parameter. We next reduce the dimensionality of each speech vector using

$$\boldsymbol{s} = \boldsymbol{P}_a^T(\boldsymbol{a} - \bar{\boldsymbol{a}}) \tag{5}$$

Speech parameters $\boldsymbol{s}$ are then concatenated with scaled appearance parameters $\boldsymbol{c}$ giving n vectors $\boldsymbol{M}_j$ defined as

$$\boldsymbol{M}_j = \left[\boldsymbol{W}_c \boldsymbol{c}_j^T, \boldsymbol{s}_j^T\right]^T \qquad j = 1, \ldots, n \tag{6}$$

where $\boldsymbol{W}_c$ is a diagonal matrix where each element is a ratio of the eigenvariances of the speech and appearance models. This process has essentially converted our clusters of appearance parameters $\boldsymbol{c}$ into new concatenated scaled speech and appearance parameters $\boldsymbol{M}$.

To linearly define these new vectors we perform PCA on each new cluster to give us k joint models of appearance and speech

$$\boldsymbol{M} = \bar{\boldsymbol{M}}_i + \boldsymbol{R}_i \boldsymbol{d} \qquad i = 1, \ldots, k \tag{7}$$

where $\bar{\boldsymbol{M}}_i$ is the mean of cluster i, $\boldsymbol{R}_i$ are the eigenvectors of cluster i and $\boldsymbol{d}$ is a speech-appearance parameter constrained to be within ±3 standard deviations from the mean of cluster i.

This new linear cluster model is essentially the heart of our synthesis model. Using statistical relationships encoded by this model we are able to estimate an appearance parameter from a speech parameter in a cluster. In the following Section we begin by describing how cluster models are chosen for synthesis before describing how appearance is estimated from speech in that cluster.

4.4 Facial area synthesis

Given an input speech signal for animating our face model we first process it using mel-cepstral analysis and perform dimensional reduction using (5). Video output for a sub-facial area is then performed on a per-frame basis. Given the speech vector for our current frame we choose which cluster in our speech-appearance model can best synthesise the facial area by finding the smallest Mahalanobis distance to the speech part of the centres of each cluster using

$$D = (\boldsymbol{s}_{input} - \bar{\boldsymbol{s}}_i)^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{s}_{input} - \bar{\boldsymbol{s}}_i) \tag{8}$$

where $\bar{\boldsymbol{s}}_i$ is the mean speech parameter in cluster i, $\boldsymbol{s}_{input}$ is our input speech parameter and $\boldsymbol{\Sigma}$ is the covariance matrix of the speech parameter training set.

Given an appropriate cluster model, we may now estimate $\boldsymbol{c}$ using $\boldsymbol{s}_{input}$. To estimate $\boldsymbol{c}$ utilise the linear description of the information in our chosen cluster. Each eigenvector in the cluster model represents a linear relationship between parameters. Projection onto this linear axis through the distribution then allows us to estimate one parameter from another. To illustrate this consider the example in Fig. 5 where we have modelled a cluster

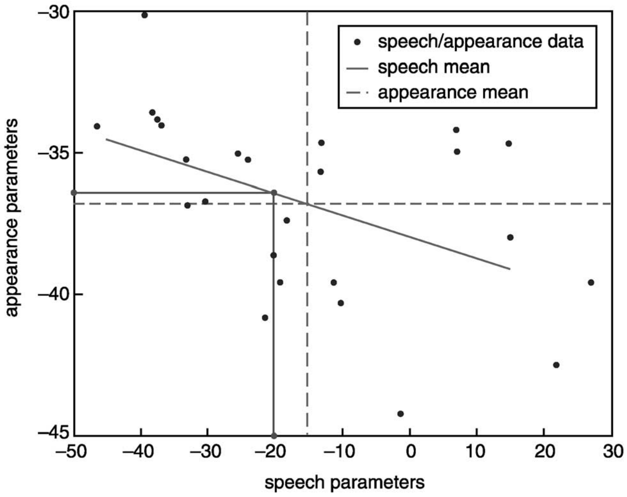

Fig. 5 *Estimation of a single-valued parameter from another using a PCA model*

of appearance and speech parameters. In this example, speech and appearance parameters are represented by single values, although the technique scales to higher-dimensional parameters. The line through the distribution shows the eigenvector associated with the largest variation. Given a speech parameter we can project onto the eigenvector and estimate an appearance parameter. To achieve this estimation-via-projection method with a linear PCA model we split the eigenvectors of a distribution into two new matrices, where each matrix contains eigenvector information for speech and appearance, respectively.

Therefore, given a cluster i, chosen using (8), we split its matrix of eigenvectors $\boldsymbol{R}_i$ into two parts, where the top part corresponds to appearance and the bottom part to speech. We then denote the linear relationship between speech and appearance in each cluster as

$$\boldsymbol{W}_c\boldsymbol{c} = \bar{\boldsymbol{c}}_i + \boldsymbol{R}_{c,i}\boldsymbol{d} \qquad (9)$$

$$\boldsymbol{s} = \bar{\boldsymbol{s}}_i + \boldsymbol{R}_{s,i}\boldsymbol{d} \qquad (10)$$

where $\bar{\boldsymbol{c}}_i$ and $\bar{\boldsymbol{s}}_i$ are the mean appearance and speech parameters of cluster i and $\boldsymbol{R}_{c,i}$ and $\boldsymbol{R}_{s,i}$ are those parts of the eigenvectors of $\boldsymbol{R}_i$ associated with appearance and speech, respectively. The vector $\boldsymbol{d}$ may be thought of as the distance along the direction of an eigenvector through a distribution (see Fig. 5). If an eigenvector approximates the distribution well enough then by calculating $\boldsymbol{d}$ we may accurately estimate values of the vectors in the distribution. Given $\boldsymbol{s}_{input}$ we calculate $\boldsymbol{d}$ using

$$\boldsymbol{d} = \boldsymbol{R}^T_{s,i}(\boldsymbol{s}_{input} - \bar{\boldsymbol{s}}_i) \qquad (11)$$

and use $\boldsymbol{d}$ in (9) to calculate $\boldsymbol{c}$. We then constrain $\boldsymbol{c}$ to be within ±3 standard deviations from the mean of its respective cluster and calculate shape $\boldsymbol{x}$ and texture $\boldsymbol{g}$ using (1) and (2). As a final step we perform a local smoothing of the constructed $\boldsymbol{x}$ and $\boldsymbol{g}$ vectors. This smoothing achieves two things: it removes noise carried into the synthesis model from the training set which would degrade reconstruction quality, and compensates for any badly chosen cluster synthesis models. This last point is important and is discussed further in Section 7.

5 Adding sub-facial areas for output

Reconstruction for output requires synthesised texture and shape information be generated for each node in the hierarchy. The important issue when adding together the different facial areas is seamless blending between them. The obvious approach is to warp the textures onto one another. However, since each facial area is driven by a separate appearance model, in which all training vectors textures are normalised with respect to each other, we discover a problem with denormalising these different textures while still maintaining a convincing blend, e.g. a mouth image may be brighter than the face image it is being added to.

The approach we use for adding the images is to estimate facial texture and shape information (i.e. information at the root of the model) which contains all of the detail in the sub-facial images (or lower nodes of the hierarchy). We have therefore redefined the problem from that of adding images to estimating a face image to match all of the sub-facial parts.

The problem is solved in two stages, one for shape and one for texture. The procedure for shape is as follows: given a shape vector for a synthesised node, we first subtract the mean shape for that node, yielding shape offsets. We then take these offsets and apply them to the mean shape of its parent node. This process is repeated from leaf nodes upwards until the root node is reached. Shape information for lower level nodes also has precedent over shape information in its parent. For example, mouth shape information synthesised for a jaw node would be replaced with information from the mouth node.

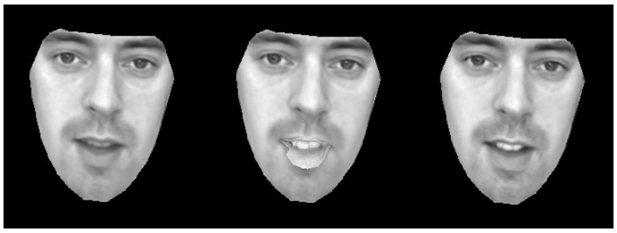

Fig. 6 *Facial texture construction (applied to adding a mouth image to a facial image)*

Given a parent image (left) we warp a child image over it into its correct position - giving a rough estimate output image (middle). After projection in and out of a parents texture PCA model a composite image is constructed (right) with no visible blends or normalisation artefacts

To add texture vectors we do the following: we warp each texture to the corresponding mean shape in its parent node, providing a rough estimate output texture. We next convert our estimate into reduced dimensional form using the texture PCA model of the parent facial area, e.g. when adding a mouth image to a jaw image we create a rough estimate and then project this through the jaw PCA model. By projecting back out of the PCA model we then synthesise a texture which contains the appearance of both sub-facial areas seamlessly integrated onto one another. Figure 6 gives an example of the reconstruction process applied to adding a mouth image to a facial image. Owing to the fact that the parent node approximates the child node we call this process ‘parent-approximation’.

To finish processing of an output frame we add colour information. Recall that so far we have only modelled luminance (or grey-level) information in our animations and that hue and saturation information for each training image has been stored. To add colour we simply find the appearance parameter in the training set which best matches our output appearance parameter and use the corresponding hue and saturation information. The luminance, hue and saturation face images are then finally warped from their mean shape to the new synthesised shape co-ordinates.

6 Evaluation

To build our synthesis model we recorded video and audio data of a subject speaking a list of words chosen to target specific visemes listed in Table 1 [21]. We labelled each of the frames with the aid of a bootstrapped active shape model (ASM). The bootstrapping approach involves manually labelling an initial set of frames to initialise the ASM and using the ASM to automatically label consecutive frames. Automatically labelled frames are then added to the ASM, increasing its robustness as the number of landmarked frames increases. Using these methods resulted in approximately 700 video frames for building our model. We next constructed a hierarchical model with a root node for the whole face and a sub-node for the mouth.

For evaluation we recorded the training subject along with five new subjects, three male and two female, speaking a list of new words targeting visemes, different numbers and the alphabet. Using this new audio, we then synthesised video-realistic reconstructions using our hierarchical model.

Table 1: Training vocabulary

Word spoken	Mouth posture name
'we'	narrow
'fit', 'heavy'	lip to teeth
'mock', 'bin', 'spark', 'pin'	lips shut
'moon'	puckered lips
'we', 'wharf', 'good'	small oval
'farm'	widely open
'up', 'upon', 'fur'	elliptical
'free'	elliptical puckered
'pit', 'believe', 'youth'	relaxed narrow
'thin', 'then'	front tongue
'time', 'dime', 'nine'	tongue to gum
'leaf'	tip to gum
'all', 'form'	spherical triangle
'see', 'zebra'	tightened narrow
'mat'	large oval
'ship', 'measure', 'chip', 'jam', 'gentle'	protrusion
'let', 'care'	medium oval
'keep', 'go', 'rank', 'rang'	undefined open

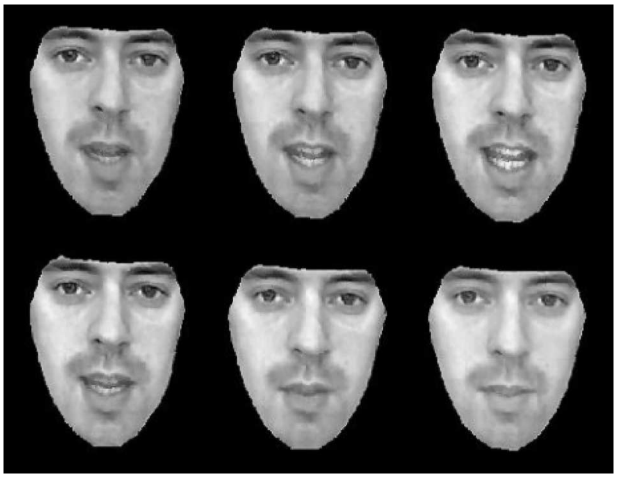

Fig. 7 *Frames from a reconstruction of a viseme-targeted word using the training subject speaker*

View from left to right, top to bottom

Figures 7 and 8 show a selection of frames from reconstructions using speech from the training subject and speech from a new subject. Both reconstructions are for words not included in the original training set. Figures 9 and 10 show trajectories of synthesised output shape and texture parameters against ground truth parameters. The synthetic trajectories are calculated using new input audio recorded from the training speaker reciting a list of viseme-targeted words not present in the original training set. The Figures show how the synthetic trajectories for new audio follow the same general pattern as the ground truth trajectories. Differences which do occur between the two signals appear in terms of signal amplitude and 'noise'. We found that differences in amplitude tended not to directly affect the perceived accuracy of the lip-synch, or the textural quality of the output images. This is because all parameter values are within a stable bound of ±2 standard deviations from the mean. Similarly noise in the parameter trajectories did not appear to be visible as an artifact in the animations. This noise may be attributed to occasional incorrect cluster choices, which in turn are due to our lack of coarticulation modelling. We discuss this further in the following Section, and propose solutions.

For the training subject and three of the new speakers we found that the animations were of a high quality, yielding strong lip-synch with the new audio irrespective of whether

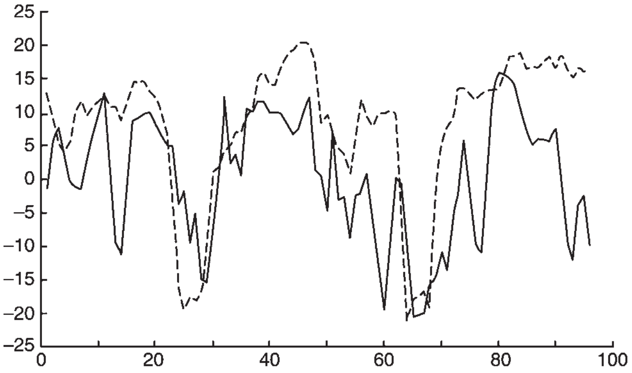

Fig. 9 *Synthetic (solid line) against ground truth (dashed line) texture PCA parameter trajectories for new audio of the training subject speaking a list of viseme-targeted words*

The texture parameter associated with the highest mode of variation is shown

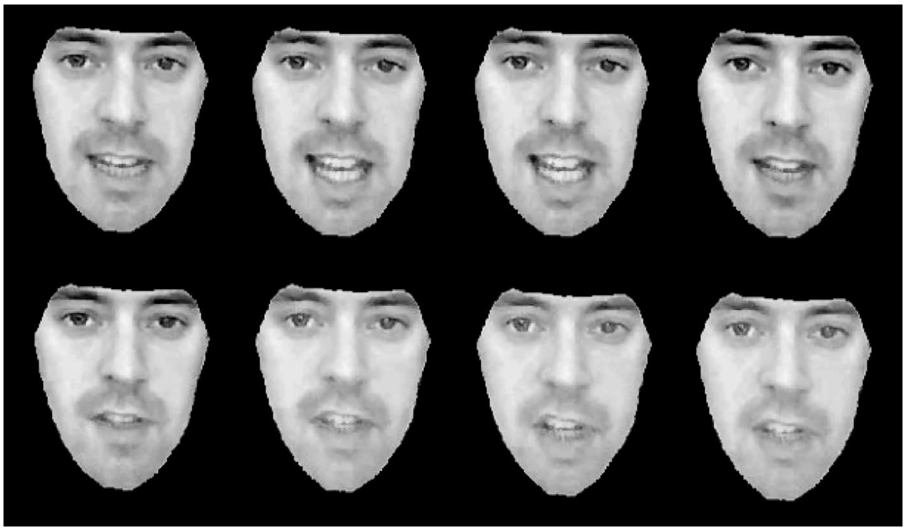

Fig. 8 *Frames from a reconstruction of letters from the alphabet using a new speaker*

View from left to right, top to bottom

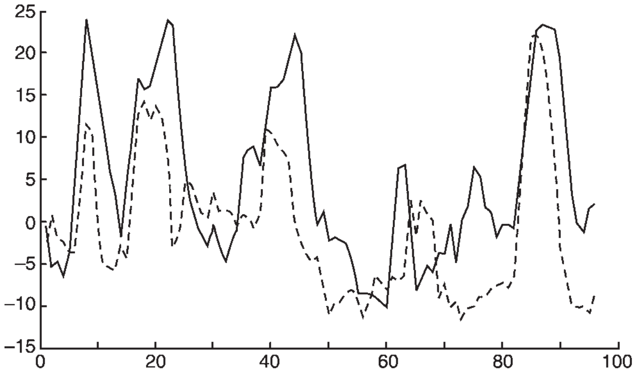

Fig. 10 *Synthetic (solid line) against ground truth (dashed line) texture PCA parameter trajectories for new audio of the training subject speaking a list of viseme-targeted words*

The texture parameter associated with the second highest mode of variation is shown

the speaker was male or female. However, animations with the two other new speakers proved less convincing. We believe that the reason for this is due to the distribution of the input speech parameters. For animations showing strong lip-synch to the audio the distributions were quite similar to that of the training subject's distribution. The distributions for the less convincing animations appeared to be different in terms of shape and scale. Figure 11 shows two speaker distributions compared to the training subject distribution. The top plot yields a poor animation while the bottom plot produces an animation with a strong lip-synch. Note how the new speaker distribution on the bottom plot is offset from the training subject's distribution, while both distributions on the top plot are more closely matched.

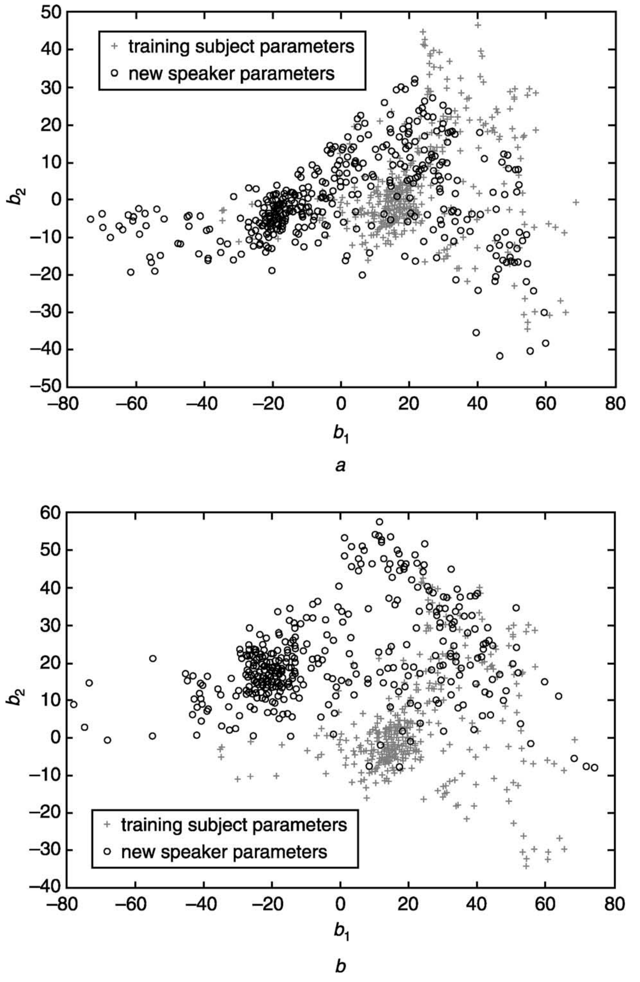

Fig. 11 *Training subject speech distribution against new speaker distributions*

a Good animation
b Poor animation

7 Discussion

We found that the major factor in contributing to the quality of lip-synch in our animations was accent. Given a similar accent to the one in the training set the animations show strong lip-synch to the input audio. We believe that the reason for the effect of accent appears to lie in the shape of the input speech distribution, which is offset from the training speakers in reduced quality animations. We intend to examine this further with the aim of transforming input speech distributions to better match the training distribution.

One animation characteristic which is not accounted for in our model is that of coarticulation. Although this does not affect short animations with normally paced speech, we have discovered that with fast speech our animations degrade. The reason for this artifact is due to inappropriate cluster choices often made during synthesis. This in turn is an artifact of the short-term speech analysis performed for synthesis, i.e. construction of an output frame independently from previous or next frames. This is essentially the coarticulation problem manifest within the system. The need for a coarticulation model is caused by the many-to-many relationship between speech and appearance (or speech and the mouth). Systems which employ phonetic inputs inherently provide a greater constraint to this problem since we are aware of which phoneme will occur next, and may guide a coarticulation algorithm towards preparation for output. With continuous speech systems such as ours, we consider the solution to the coarticulation problem as being correct cluster choice at time *t*. It is our intention in future work to model cluster choice directly using a time-series model such as a HMM. We have already discovered with some initial testing that given a correct cluster choice we can produce animations which are perceptually indistinguishable from real animations given speech of any tempo.

As well as mouth animation in our output videos we also notice some non-verbal animations, such as facial expression. This is exhibited as different facial mannerisms appearing in the animations associated with the unique style of the subject modelled in training. We attribute this effect to the modelling of speech continuously and is an advantage often seen in facial animation systems of this type [8]. In future work we would like to model this phenomenon more specifically in order to fully exploit facial behaviour.

8 Conclusions

We have introduced a nonlinear hierarchical speech-appearance model of the face capable of producing high-quality realistic animation given a speech input. The model is capable of synthesising convincing animation given new audio from either the training speaker or a new speaker. The system is also purely data driven (using continuous speech signals) requiring no phonetic alignment of speech before video synthesis. In future work we hope to extend the model by encoding relations between sub-facial areas and emotional content derived from speech. We also plan to improve certain areas of synthesised coarticulation with the inclusion of a time-series-based model.

9 References

1 Parke, F.: 'Computer generated animation of faces'. Proc. ACM National Conf., 1972

2 Kalberer, G.A., and Van Gool, L.: 'Realistic face animation for speech', *J. Vis. Comput. Animat.*, 2002, **13**, pp. 97–106

3 Reveret, L., Bailly, G., and Badin, P.: 'Mother: a new generation of talking heads providing a flexible articulatory control for video-realistic speech animation'. Proc. 6th Int. Conf. on Spoken Language Processing (ICSLP), Beijing, China, Oct. 2000

4 Ezzat, T., Geiger, G., and Poggio, T.: 'Trainable videorealistic speech animation'. Proc. Computer Graphics and Interactive Techniques, San Antonio, TX, USA, July 2002, pp. 388–398

5 Theobald, B., Cawley, G., Glauert, J., and Bangham, A.: '2.5 d visual speech synthesis using appearance models'. Proc. BMVC, Norwich, UK, 2003, vol. 1, pp. 43–52

6 Bregler, C., Covell, M., and Slaney, M.: 'Video rewrite: driving visual speech with audio'. Proc. 24th Conf. on Computer Graphics and Interactive Techniques, 1997, pp. 353–360

7 Le Goff, B., and Benoit, G.: 'A text-to-audiovisual-speech synthesiser for French'. Proc. Int. Conf. on Spoken Language Processing (ICSLP), 1996

8 Brand, M.: 'Voice puppetry'. Proc. Computer Graphics and Interactive Techniques, 1999, pp. 21–28

9 Beskow, J.: 'Rule-based visual speech synthesis'. Proc. Eurospeech, 1995, pp. 299–302

10 Huang, F.J., and Chen, T.: 'Real-time lip-synch face animation driven by a human voice'. Proc. IEEE Workshop on Multimedia Signal Processing, Los Angeles, CA, USA, 1998

11 Hong, P., Wen, Z., and Huang, T.S.: 'Real-time speech-driven face animation with expressions using neural networks', *IEEE Trans. Neural Netw.*, 2002, **13**, (4), pp. 916–927

12 'Final fantasy'. DVD edition, Columbia Tri-Star, 2001

13 Cosatto, E., and Graf, H.P.: 'Photo-realistic talking-heads from image samples', *IEEE Trans. Multimed.*, 2000, **2**, (3), pp. 152–163

14 Karaulova, Y., Hall, P., and Marshall, A.D.: 'Tracking people in three dimensions using a hierarchical model of dynamics', *Image Vis. Comput.*, 2002, **20**, (9-10), pp. 691–700

15 Cootes, T.F., Taylor, C.J., Cooper, D.H., and Graham, J.: 'Active shape models: their training and application', *Comput. Vis. Image Underst.*, 1995, **61**, (1), pp. 38–59

16 Cootes, T., Edwards, G., and Taylor, C.: 'Active appearance models', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2001, **23**, (6), pp. 681–684

17 Luettin, J., and Thacker, N.A.: 'Speechreading using probabilistic models', *Comput. Vis. Image Underst.*, 1997, **65**, (2), pp. 163–178

18 Deller, J., Proakis, J., and Hansen, J.: 'Discrete-time processing of speech signals' (1993)

19 Cootes, T., and Taylor, C.: 'A mixture model for representing shape variation'. Proc. British Machine Vision Conf., 1997, pp. 110–119

20 Bowden, R.: 'Learning non-linear models of shape and motion'. PhD thesis, Dept. of Systems Engineering, Brunel University, UK, 2000

21 Nitchie, E.: 'How to read lips for fun and profit' (Hawthorn Books, New York, USA, 1979)