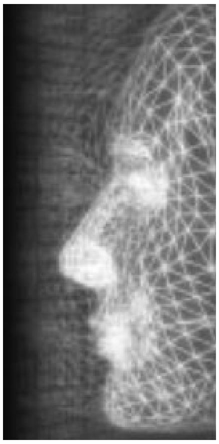
COMPUTER ANIMATION AND VIRTUAL WORLDS
Comp. Anim. Virtual Worlds 2010; **21**: 343–354
Published online 26 May 2010 in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/cav.340

Automatic construction of 3D animatable facial avatars



By Yujian Gao, Qinping Zhao, Aimin Hao, T. M. Sezgin and N. A. Dodgson*

Rigging for facial animation is an important but time-consuming task, which generally requires experienced artists with knowledge of facial anatomy. In this paper, we investigate whether it is possible to produce a good animatable avatar automatically, given only a 3D static triangle mesh of the head. An automatic mechanism is devised for constructing multi-layer animatable facial avatars for unseen faces. We evaluate our technique with a variety of models, and give a quantitative analysis of the constructed results. We also designed and conducted a user study for evaluating the perceived quality of the generated expressive animations. The results demonstrate that our method is an appropriate tool for naïve users to customize their personal 3D avatars. Copyright © 2010 John Wiley & Sons, Ltd.

KEY WORDS: facial animation; avatar; multi-layer model; landmark detection; rigging

Introduction

Since Parke's pioneering work in the early 1970s [1], facial animation has been widely used in entertainment, virtual environment and low bandwidth teleconferencing. In some professional applications such as films or virtual newscasters, high quality photorealistic facial animations are usually achieved at the cost of intensive manual rigging and tuning by highly trained artists with knowledge of facial anatomy. But this is not the main focus of our paper, and high fidelity is not the only measuring stick for facial animation. There are certain popular applications where user-level control and customization are more important than having complex photorealistic models. For example, in virtual environments or online games where users have lookalike avatars of themselves, it is vital for users to be able to customize their avatars easily and quickly. However, most of the current animation techniques lack universality and require intensive manual rigging from scratch when facial geometry changes. This results in a bottleneck in facial animation applications. Therefore, an emerging interesting question is: is it possible to produce a good animatable model of a head automatically, given only a 3D static triangle mesh of the head?

Motivated by this question, we investigate and compare the current facial animation techniques, and present an automatic method for constructing multi-layer animatable facial avatars. Our work provides a workable mechanism by which the muscles and additional animation controls can be positioned automatically. The entire construction process is depicted in Figure 1. Our method consists of two steps: first, we detect 3D facial landmarks using a novel method which combines both 2D and 3D information. Second, a predictor is trained by learning the mapping between landmarks and underlying muscle positions, and then used for muscle construction. Our method makes it straightforward for naïve users to generate animatable faces and customize their own avatars without expert knowledge.

The main contributions presented in this paper are:

- A new mechanism for landmark detection on 3D facial meshes. Our method makes use of both 2D and 3D information for robust detection, in particular improving over previous 3D shape analysis methods. It can also be used as a pre-processing step for many techniques such as mesh parameterization or feature-based interpolations.

*Correspondence to: Y. Gao, State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, 83 Xueyuan Road, Beijing, China.
E-mail: gaoyj@vrlab.buaa.edu.cn

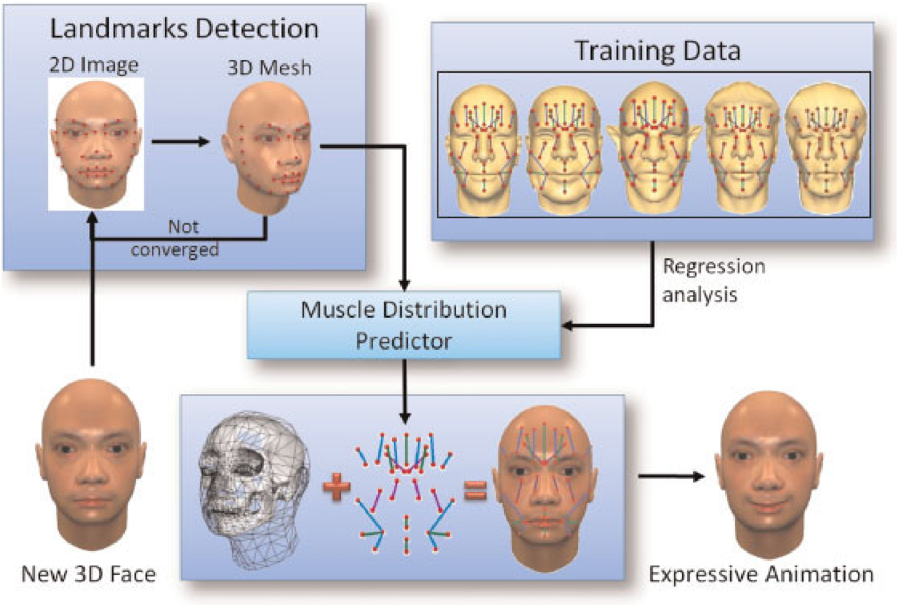


Figure 1. The construction process of an animatable facial avatar.

- A novel method for automatically constructing multi-layer animatable facial avatars. We employ regression techniques to investigate how well the automatically-captured landmarks can be mapped to an underlying muscle distribution. A predictor is trained to predict muscles from detected 3D landmarks.

The next two sections describe our algorithms for 3D landmark localization and multi-layer model construction. We then present the results of our evaluation in experiment section, where we demonstrate that our system generates acceptable animations for a wide range of 3D facial models. We finally discuss related work in the light of our new method.

3D Facial Landmark Detection

Most of the previous landmark detection methods focus on the shape and curvature features of 3D meshes.[2–5] However, they all suffer from problems caused by surface irregularities of models, and also depend on a pre-processing step that identifies which part of the model is face. Besides, curvature-based methods are only capable of detecting landmarks with distinct curvature features, i.e. eye corners and mouth corners, which are insufficient for our muscle prediction task. Motivated by 2D feature point detection techniques, we incorporate texture information into our landmark detection method. By doing this, we avoided running into the limitations inherent in techniques that use only curvature information.

To utilize both the 2D and 3D information for robust landmark detection, we follow an iterative search mechanism. First, an image-based landmark detection method is devised to obtain an initial estimate of landmark locations. This step mainly focuses on exploiting the information conveyed by 2D texture and guaranteeing the global shape as well as the spatial relativity of landmarks. The second step is to refine the location of landmarks according to the local curvature within constrained areas. We iteratively apply these two steps until the results converge or a maximal iteration reached.

Image-based Landmark Detection

The textures of 3D models are usually distorted or separated when they are flattened, therefore we cannot directly use them for 2D feature point detection. In order to prepare smooth texture data, we first render a frontal image of the face model with texture under uniform illumination. We assume that the face is oriented towards the *z*-axis. This is the default orientation for the majority of 3D facial models. For the exceptional

DOI: 10.1002/cav

cases, the proper orientation can be obtained through principal component analysis (PCA) or by manual adjustment. The rendered image is then fed into a 2D facial feature detector implemented based on Bayesian Tangent Shape Model (BTSM).[6] Considering that the muscles are scattered all over the face, using only the landmarks around eyes and mouth will dramatically degrade the prediction quality, therefore we set more landmarks to cover the whole face including boundary of the frontal face (see Figure 2).

After locating the 2D feature points, a 2D to 3D projection is applied to compute the corresponding landmarks on the 3D mesh. This is achieved by projecting the landmarks perpendicular to the image towards the model (see Figure 3a). The point where each projection ray hits the 3D face surface is preserved as the initial landmark location. This image-based landmark detection method does not take 3D geometry into account, therefore it is invariant to the complexity of the 3D model, and it works effectively even for substantially complex 3D models which contain other objects with face-like curvature features.

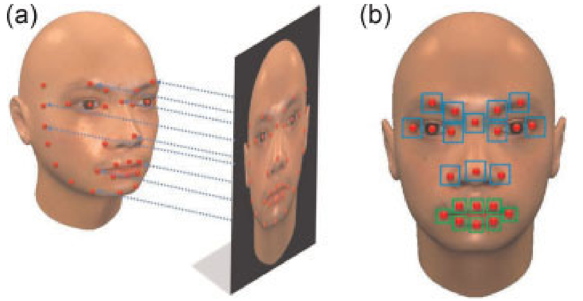


Figure 3. (a) The 2D–3D projection scheme. (b) Search areas for landmark refinement.

Landmark Refinement using Shape Analysis

Although human faces vary a lot due to sex, race, etc., certain regions of the face have characteristic curvature signatures. Therefore, these intrinsic curvatures can be used for conducting the refinement of landmark locations. Previous works[2–5] have shown that if the search areas can be limited to reasonably small regions, the accuracy of search results could be very high. Given the initial landmarks obtained in the previous step, we can easily confine the first 22 landmarks' refinement process to rectangular regions centred at the current positions, see Figure 3b (the boundary landmarks remain unchanged in this step). We define the side length of the search areas as:

$$L_{\text{mouth}} = \frac{\| \overleftarrow{P_{13}P_{15}} \| + \| \overleftarrow{P_{14}P_{16}} \|}{2}, L_{\text{eye}} = \frac{\| \overleftarrow{P_{17}P_{1}} \| + \| \overleftarrow{P_{18}P_{2}} \|}{2}. \quad (1)$$

where L_{mouth} is the search length for mouth areas (green squares), and L_{eye} is for areas around eyes (blue squares). These search lengths constrain the search results to be local optimum.

Each landmark has a curvature type (see Table 1). Once the search direction is decided, we estimate the Gaussian (K) and the mean (H) curvatures for points within the search areas by computing the partial derivatives. Using these two curvature values, we determine the curvature type of each point based on Table 2. Of all the landmark candidates within each search area, the point, which has the most obvious corresponding curvature feature in Table 1, is chosen as the new landmark.

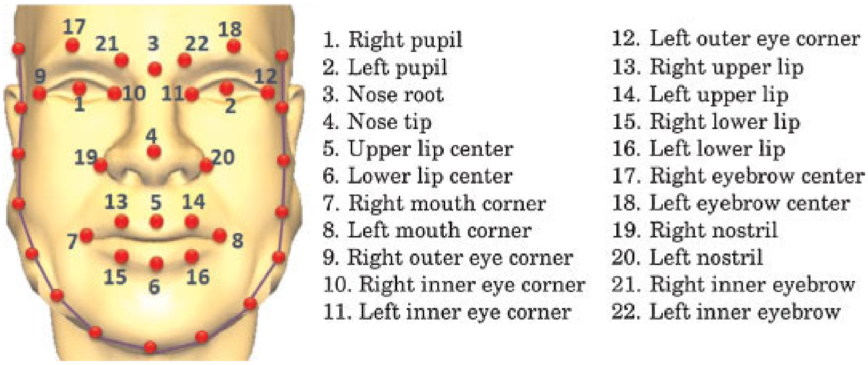


Figure 2. The 22 feature points on the face and 15 feature points on the facial outline.

FP3	FP4	FP5	FP6	FP7
saddle ridge	nose tip	peak	peak	pit
FP8	FP9	FP10	FP11	FP12
pit	pit	pit	pit	pit
FP13	FP14	FP15	FP16	FP17
peak	peak	peak	peak	peak
FP18	FP19	FP20	FP21	FP22
peak	pit	pit	peak	peak

Table 1. Curvature types of different feature points.

When this step finishes, we check whether the convergence has been achieved by comparing current results with previous iteration. If not, the landmarks would then be projected back onto the 2D image, and the image-based landmark localization will be carried out again.

Automatic Model Rigging

We employ Zhang's[7] nonlinear multi-layer architecture as the rigging model. Justification for choosing this model is given in Section 5. Note that our contribution is not the multi-layer model but the automatic mechanism of generating animatable avatars. To rig the face model, our first task is predicting muscles using the detected landmarks. Therefore, we need to train a predictor that learns the spatial relationship between the landmark positions and the muscles from examples. Since we cannot determine beforehand whether the mapping is linear or nonlinear, we tried both of them and made an objective comparison between their results.

Training Data Collection

Before training, we first collected a set of 50 3D triangulated facial models as training data, covering a wide range of human facial variation. All these models were in Wavefront .obj format and normalized into the same coordinate system, with their geometric centre at the origin. Three-dimensional landmarks were localized for each model automatically with our proposed method, whereas muscles were constructed and adjusted manually for each face: an initial coarse muscle model was first built for each facial geometry, which was then manually tuned by artists experienced in facial animation until plausible expressive animations were obtained. Although the process of building facial muscles is labour-intensive, it only needs to be done once. For all subsequent unseen faces, we only need the mapping previously learned from the training data to construct the muscles. Figure 4 shows some of the training models with landmarks and muscles attached.

	$K < 0$	$K = 0$	$K > 0$
$H < 0$	Saddle Ridge	Ridge	Peak
$H = 0$	Minimal	Flat	(None)
$H > 0$	Saddle Valley	Valley	Pit

Table 2. Curvature classification based on HK map components.

CCA-based Linear Regression

The predictability between variables greatly depends on their inter-correlation, that is, how well we can estimate the muscles from the landmarks depends on how great an inter-correlation exists between them. Canonical correlation analysis (CCA) is a statistical technique developed by Hotelling[8] for revealing the functional dependencies between two sets of measurements. It can maximize the correlation between the two sets of variables, and is well-suited as a pre-processing step for regression because of its ability to capture data dependency while avoiding overfitting.

Each of the 23 linear muscles and sheet muscles is determined by two endpoints, therefore all the muscles of one face are determined by 46 points. We define $n = 50$ as the number of training examples, $p = 37$ as the number of landmarks and $q = 46$ as the number of muscle endpoints on each face. Then the training data can be expressed using two sets of variables, denoted as $L = (l_1, l_2, \ldots, l_p)^{\mathrm{T}}$ and $M = (m_1, m_2, \ldots, m_q)^{\mathrm{T}}$. Note that l_i and m_i are both n-vectors containing all the 50 measurements of ith landmark and muscle endpoint, respectively.

The CCA scheme requires that the covariance matrices of L and M be of full rank. To remove multicollinearities and avoid the inversion of non-full rank matrices, we first perform PCA on L and M which are then represented as:

$$L = E_L \cdot X \quad \text{and} \quad M = E_M \cdot Y, \tag{2}$$

where E_L and E_M are the eigenvector matrices, X and Y correspond to the low-dimensional principal data on which CCA will be performed.

DOI: 10.1002/cav

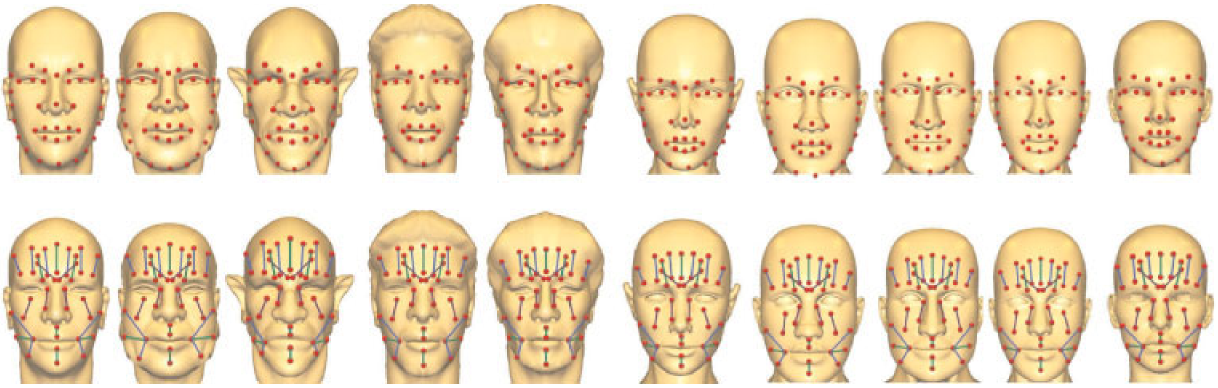


Figure 4. A portion of the training data with landmarks and muscles labelled.

CCA extracts the correlated modes between X and Y by seeking a set of vector pairs A_i and B_i, which yields the canonical variates u_i and v_i with maximum correlation:

$$u_i = X^{\mathrm{T}} A_i \quad \text{and} \quad v_i = Y^{\mathrm{T}} B_i. \tag{3}$$

The correlation between u_i and v_i can be expressed as:

$$\rho_i = \frac{A_i^{\mathrm{T}} C_{XY} B_i}{\sqrt{A_i^{\mathrm{T}} C_{XX} A_i B_i^{\mathrm{T}} C_{YY} B_i}}, \tag{4}$$

where C_{XY} is the cross-covariance matrix of X and Y, C_{XX} and C_{YY} are auto-covariance matrices. We then compute the partial derivatives of ρ_i with respect to A_i and B_i, respectively, and set the derivatives to be zero to maximize the correlation ρ_i. So we have

$$\begin{cases} C_{XX}^{-1} C_{XY} C_{YY}^{-1} C_{YX} A_i = \rho_i^2 A_i \\ C_{YY}^{-1} C_{YX} C_{XX}^{-1} C_{XY} B_i = \rho_i^2 B_i \end{cases} \tag{5}$$

By solving Equation (5) using singular value decomposition (SVD), we can obtain the decreasingly sorted correlations $\{\rho_1, \rho_2, \ldots, \rho_r\}$ and the corresponding transformation vectors $\overline{A} = [A_1, A_2, \ldots, A_r]$, $\overline{B} = [B_1, B_2, \ldots, B_r]$. We also get the corresponding sets of canonical variates $U = [u_1, u_2, \ldots, u_r]$ and $V = [v_1, v_2, \ldots, v_r]$ via Equation (3). Therefore, the correlation between canonical variates is maximized and hence the predictability between u_i and v_i is maximized. After the CCA transformation, a basic linear regression is then performed in the canonical space to train the predictor P which estimates v_i from u_i.

For any new face model, the whole prediction procedure is illustrated in Figure 5, where the thicker arrow denotes higher correlation and more important estimation.

Kernel CCA-based Regression

CCA might be insufficient to extract accurate descriptors of the data because of its linearity, whereas kernel CCA offers an alternative nonlinear solution. It works by mapping the original data into a higher dimensional feature space and solving a corresponding nonlinear version of the problem in that feature space. This method is known as the 'kernel trick'.

Feng *et al.*[9] once used KCCA for mesh deformation. They built a connection between the bone deformations and the movement of control points. In contrast, we use KCCA-based regression to predict muscle coordinates from landmark positions. Furthermore, we kernelized both the input landmark data and the output muscle data, and compute dual bases of CCA bases for accurate reconstruction, whereas Feng[9] kernelized only the input control points data. Given the kernelized data, the following training procedure has a similar manner as linear CCA regression. Interested readers can refer to Reference 10 for more details.

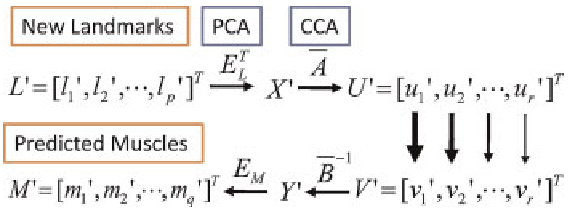

New Landmarks
PCA
CCA
$L' = [l_1', l_2', \cdots, l_p']^T \xrightarrow{E_L^T} X' \xrightarrow{\overline{A}} U' = [u_1', u_2', \cdots, u_r']^T$
Predicted Muscles
$M' = [m_1', m_2', \cdots, m_q']^T \xleftarrow{E_M} Y' \xleftarrow{\overline{B}^{-1}} V' = [v_1', v_2', \cdots, v_r']^T$

Figure 5. The prediction process for a new face.

DOI: 10.1002/cav

Sphincter Muscle and Skull Fitting

The sphincter muscle is modelled as a parametric ellipsoid. We compute its centre C as the geometrical mean of the eight landmarks around mouth. With the ith landmark denoted as $FP(i)$, the semi-major axis a and the semi-minor axis b are computed as

$$a = \frac{\| C - FP(7) \| + \| C - FP(8) \|}{2}, b = \frac{\| C - FP(5) \| + \| C - FP(6) \|}{2}, \tag{6}$$

For better visual realism, a generic skull model is automatically fitted within the facial mesh (see Figure 6) using registration technique described in Reference 11, but since we already have the facial landmarks, we can fully automate the fitting process without manual intervention. The constructed muscles' origin points are then projected onto the skull and the insertion points are attached to the original facial skin mesh.

The muscle contraction mechanism follows Zhang's[7] method. Before animating these muscles, there are several additional properties need to be computed, e.g. the maximal influence angle/radius of linear muscle and the length/width of the influence rectangle zone of sheet muscle. The maximal influence angle of linear muscle is a property that is independent of facial scale, therefore we individually adopt the mean angular value for each vector muscle from the pre-tuned models. The other properties vary across the training models, and depend mainly on the muscle length. A solution that works well is to perform linear regression for each property of each muscle with the muscle length over the 50 training examples.

Experimental Results

Prediction Accuracy

We used a set of $n = 50$ facial models which contain 4000–5000 triangles along with their hand-tuned muscle models to measure the prediction accuracy of our method. Note that the 3D landmark detection as well as the animation results can be affected by the resolution of the polygonal face mesh, therefore a minimum number of 1000 triangles is suggested for our method. We adopted a leave-one-out cross validation scheme, where at each iteration we left out one of all the facial models as our test model, and train the mapping between landmark positions and muscles using the remaining models. Then we compared the ground truth locations of the test models' muscles with the automatically constructed results. To measure the error of prediction, we define the *Relative Error* (RE) of the ith predicted muscle as:

$$\mathrm{RE}_i = \frac{\mathrm{MeanDistError(MDE)}}{\mathrm{MuscleLength(ML)}}, \tag{7}$$

$$\mathrm{MDE} = \frac{\| P'_{2i-1} - P_{2i-1} \| + \| P'_{2i} - P_{2i} \|}{2}, \tag{8}$$

$$\mathrm{ML} = \| P_{2i-1} - P_{2i} \|, \tag{9}$$

where P_{2i-1} and P_{2i} are the origin and the insertion point of the ith ground truth muscle, whereas P'_{2i-1} and P'_{2i} are of the ith predicted muscle, respectively. After each iteration, we can obtain a measurement vector $(\mathrm{RE}_1, \mathrm{RE}_2, \ldots, \mathrm{RE}_{23})$ of all the 23 automatically constructed muscles.

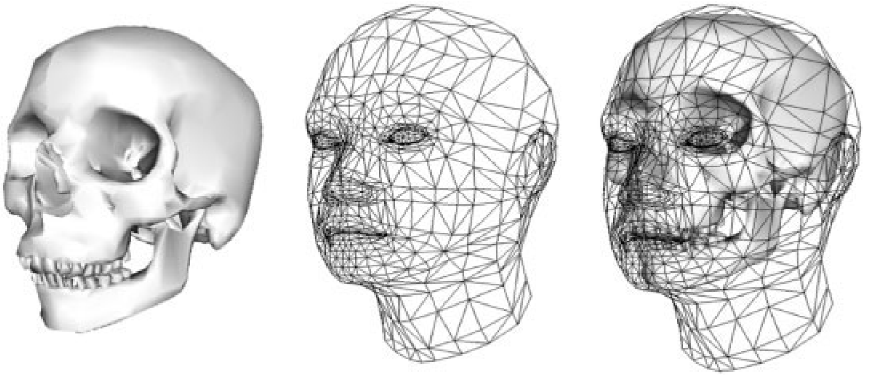


Figure 6. The generic skull before and after fitted to the facial mesh.

DOI: 10.1002/cav

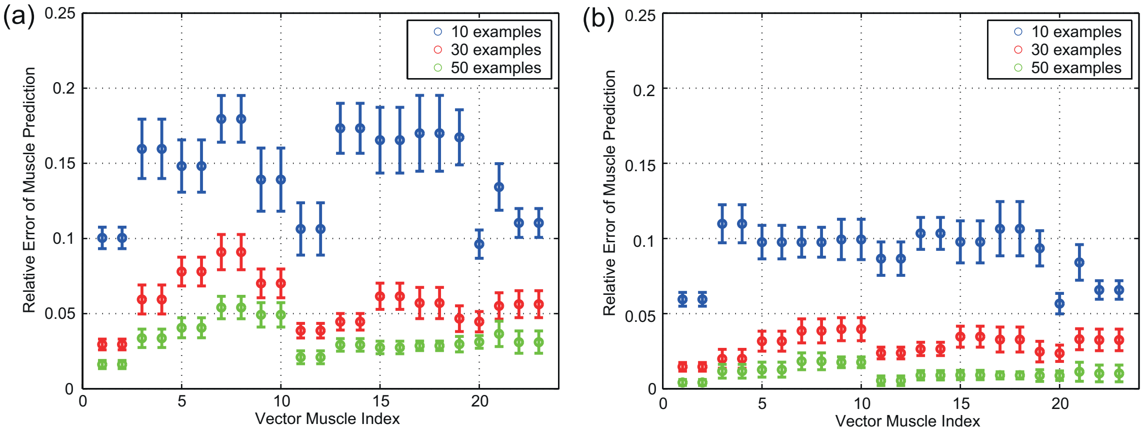


Figure 7. (a) REs of CCA-based linear regression. (b) REs of kernel CCA-based method.

To investigate how the prediction accuracy varies as the number of training examples increases, we carried out the validation scheme three times using $n = 10, 30, 50$ test examples. Figure 7 illustrates the mean values and standard deviations of $(\mathrm{RE}_1, \mathrm{RE}_2, \ldots, \mathrm{RE}_{23})$ over the n test examples using CCA-based linear regression and KCCA-based method, respectively. As we can see, KCCA-based method achieved more accurate prediction results than the linear method. As the number of training examples increased, RE converged quickly, and even as few as 30 examples resulted in small RE.

Emotional Expressiveness

We designed and conducted a user study for evaluating the degree to which the emotions in our synthesized facial animation can be recognized. Six animation scripts specifying the muscle activation data for different emotions (anger, fear, surprise, sadness, joy and disgust) were generated by artists. Note that our constructed models are not only capable of representing these six emotions, more expressions are shown in the next section. Both the hand-tuned and automatically constructed multi-layer models were used to animate each emotion on a set of 16 facial models, yielding a total of 16 models $\times$ 6 emotions $\times$ 2 sets-of-muscles $=$ 192 animation video clips.

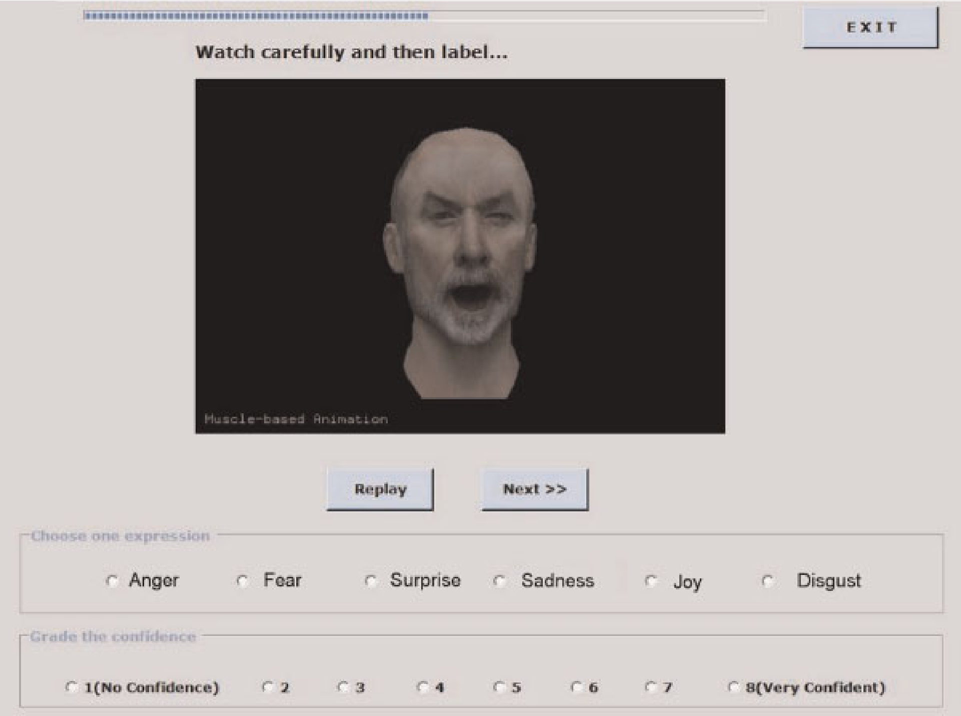
The evaluation was carried out through a computer-based interface, which allowed participants to watch randomly presented video clips and label them for six emotions. A cross-hair was displayed between consecutive video clips to fixate attention and clear the mind from previous visualization. Figure 8 shows a screenshot of the labelling interface.

Ten participants (five male, five female), aged between 20 and 34, volunteered to take part in the study, and they all have normal or corrected-to-normal vision. Each of them completed the experiment individually. We collected and analysed the participants' labelling results, as shown in Figure 9. As seen in the figure, the hand-tuned models and the constructed ones appear to give substantially identical results. We ran a signed-rank test on the labelling results, and the difference was not found to be significant. Since the experiment involved multiple raters rating multiple categories, we computed Fleiss' kappa[12] as a measure of inter-rater agreement. The results are both 'almost perfect' agreement with values of 0.824 and 0.822 for hand-tuned and predicted models, respectively.

Universality

Finally, we tested the universality of our method by applying it to models of different types, including thin and fat faces from both genders, and covering a wide range of ages, as seen in Figure 10. We also tested our approach with models that, in addition to a face, also contain objects with complex geometry in the background. The results in Figure 11 show that, because we start our processing with an appearance-based face localization step, the faces can be correctly located and animated without being confused by the background clutter.

DOI: 10.1002/cav



Watch carefully and then label...

Choose one expression

○ Anger ○ Fear ○ Surprise ○ Sadness ○ Joy ○ Disgust

Grade the confidence

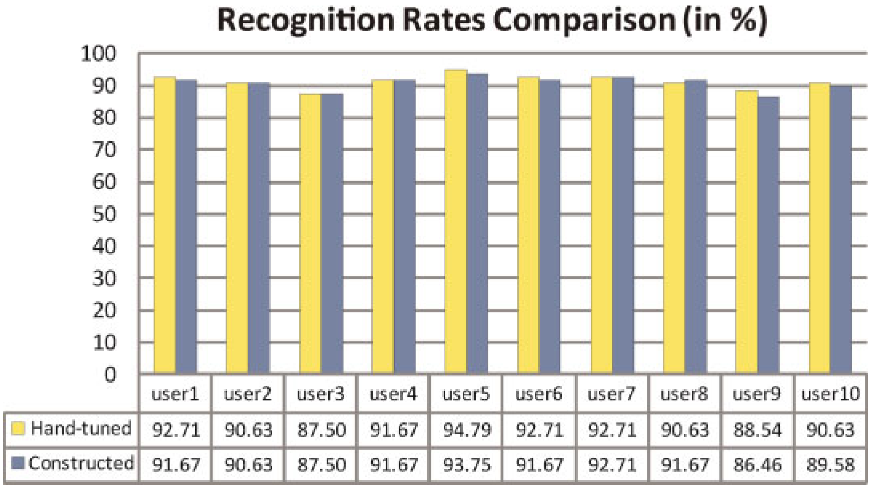
○ 1(No Confidence) ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8(Very Confident)

Figure 8. Screenshot of the labelling interface.

Related Work

3D Landmark Detection

There are a variety of facial feature detection algorithms operating on 2D colour and greyscale images, while 3D landmark localization is a relatively new area of research. Several authors have proceeded by locating the nose tip first,[13,14] and determining candidates for the remaining landmarks based on their relative locations to the nose tip. Other authors have suggested shape descriptors for landmark detection, e.g. Moccozet *et al.*[15] used the multi-scale bubbles introduced by Mortara *et al.*,[16] Chua *et al.* [17] proposed point signatures as a local descriptor, and a Gabor filter-based curvature approach has also been attempted.[18]



	user1	user2	user3	user4	user5	user6	user7	user8	user9	user10
🟨 Hand-tuned	92.71	90.63	87.50	91.67	94.79	92.71	92.71	90.63	88.54	90.63
■ Constructed	91.67	90.63	87.50	91.67	93.75	91.67	92.71	91.67	86.46	89.58

Figure 9. Recognition rates comparison.

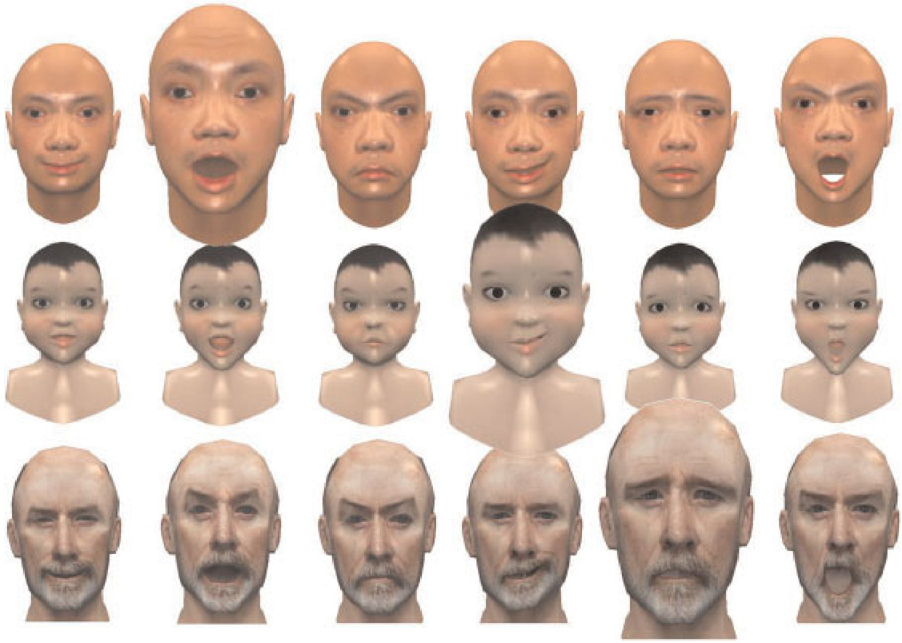


Figure 10. Expressions animated on various head models.

Curvature features are intrinsic properties of 3D models, Gordon,[2] Li *et al*.,[3] Colombo *et al*.[5] and Colbry *et al*.[4] have all developed curvature-based techniques to extract 3D facial landmarks. However, they all suffer from problems caused by surface irregularities of models. Furthermore, they depend on a pre-processing step that identifies which part of the model is the face. Our 2D landmark detection stage provides an excellent initial estimate for the true landmark locations and the curvature-based refinement step produces a better

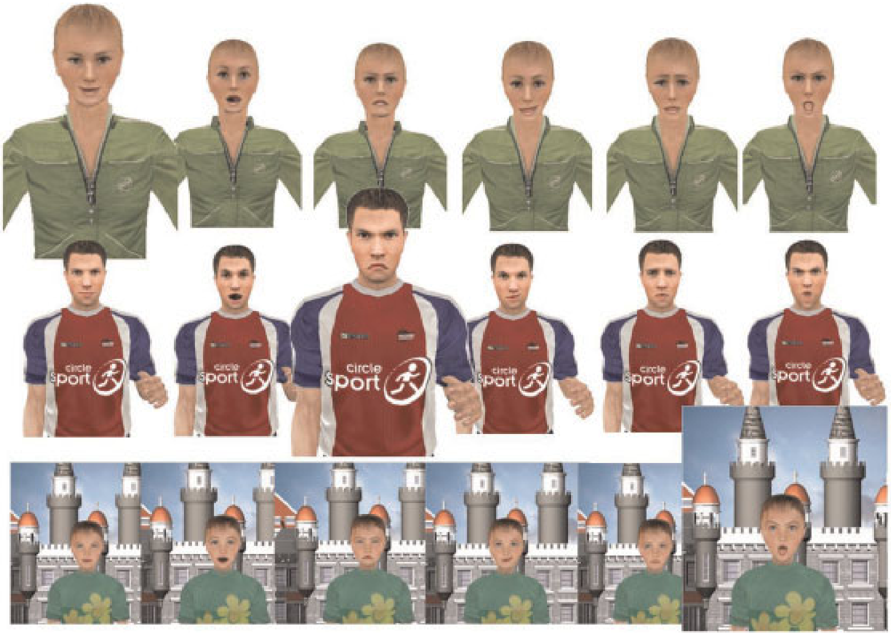


Figure 11. Expression animated on complex character models with body or castle.

DOI: 10.1002/cav

refined result without running into the limitations inherent in techniques that use only curvature information.

Simulation-based Facial Animation

According to the recent surveys,[19,20] facial animation can be roughly classified into blend shape-based approaches,[21–25] performance-driven animation[26–31] and simulation-based approaches.[7,32–38] Most approaches of the first two categories are either in lack of visual realism or in need of large data collection, therefore they are not suitable for our task. For example, early performance-driven methods[26,27] have artifacts introduced by geometrical warping operation, while recently developed methods[30,31] and all the blend shape approaches[21–24] need large collection of facial meshes with different extreme expressions to generate animation for just a single person.

In contrast, although the simulation-based approaches require substantial rigging work, it needs only one model and provides reasonable realism by simulating the anatomical structure of human face. In general, there are three major categories of simulation-based facial animation: pseudo vector muscle model, mass-spring model and layered spring muscle model. In this paper we employ Zhang's[7] nonlinear multi-layer architecture as the rigging model. This is a combination of the pseudo muscle approach with the anatomy-based facial modelling, which significantly improves the realism of synthetic facial expressions compared to the earlier techniques. Note that although anatomically accurate models have been proposed recently which claim improvement on visual realism (see Reference 38), they trade off real-time performance for accurate simulation, moreover their models are too complex to construct automatically.

Rigging

Rigging is a process analogous to setting up the strings that control a puppet's movement.[39] The traditional animation pipeline requires each character to be rigged manually, making it difficult to reuse the same rig on different characters. Recently, several methods have been devised for simplifying the tedious and labour-intensive rigging process.

Kähler *et al.*[40] devised an editing interface for artists to interactively specify muscles on 3D face geometry. Unfortunately, expert anatomical knowledge is still required for using this tool, whereas our goal is to automate this procedure entirely. Orvalho *et al.*[41] introduced a facial deformation system that reduce artists' effort of rigging facial models from scratch. But it needs the artists to label the landmarks manually and uses a sophisticated facial rig as source model. This method sacrifices full automation to achieve more realistic facial animation for professional applications like films, while we trade off some realism for full automation without the need of manual intervention or source rigged models.

Conclusions and Future Work

Expressive facial expression can be synthesized through several animation techniques, but manual rigging for facial animation is always time-consuming and labour-intensive. In this paper, we addressed this problem by automatically detecting 3D facial landmarks and learning the mapping between landmarks and the underlying muscles of the face. Our solution avoids the painstaking manual rigging process which requires knowledge of human facial anatomy. We show the usability of our framework by building a system that allows novice users to generate animatable avatars from 3D raw facial geometry. Our method can be used in a wide range of applications for common users such as in-game avatars, chat room agents, virtual environment characters, etc. Furthermore, our experiments have demonstrated that our automatic facial avatar construction method is an appropriate solution for those applications. Finally, we believe that automatically constructing animatable cartoon characters would be an interesting extension to this work. Unlike the human face, cartoon characters always deviate from the rules of facial anatomy, therefore simulation-based animation may not be appropriate anymore and other animation schemes should be developed to tackle this problem.

References

1. Parke F. Computer generated animation of faces. In *Proceedings of ACM'72 Annual Conference*, 1972; 451–457.
2. Gordon GG. Face recognition based on depth and curvature features. In *Proceedings of CVPR '92*, 1992; 808–810.
3. Li P, Corner BD, Paquette S. Automatic landmark extraction from three-dimensional head scan data. In *Proceedings of SPIE '02*, 2002; vol. 4661: 169–176.
4. Colbry D, Stockman G, Jain AK. Detection of anchor points for 3d face verification. In *Proceedings of CVPR '05*, 2005; 118.

DOI: 10.1002/cav

5. Colombo A, Cusano C, Schettini R. 3d face detection using curvature analysis. *Pattern Recognition 2006*; **39**(3): 444–455.
6. Zhou Y, Gu L, Zhang HJ. Bayesian tangent shape model: estimating shape and pose parameters via bayesian inference. In *Proceedings of CVPR '03*, 2003; vol. 1: 109.
7. Zhang Y, Prakash EC, Sung E. A new physical model with multilayer architecture for facial expression animation using dynamic adaptive mesh. *IEEE Transactions on Visualization and Computer Graphics (TVCG) 2004*; **10**(3): 339–352.
8. Hotelling H. Relations between two sets of variates. *Biometrika 1936*; **280**(3/4): 321–377.
9. Feng W, Kim B, Yu Y. Real-time data driven deformation using kernel canonical correlation analysis. In *Proceedings of SIGGRAPH '08*, 2008; pages 1–9.
10. Hardoon DR, Szedmak SR, Shawe-taylor JR. Canonical correlation analysis: an overview with application to learning methods. *Neural Computation 2004*; **16**(12): 2639–2664.
11. Aina OO. Generating anatomical substructures for physically-based facial animation. *The Visual Computer 2009*; **25**(5–7): 617–625.
12. Fleiss JL. Measuring nominal scale agreement among many raters. *Psychological Bulletin 1971*; **76**(5): 378–382.
13. Xu C, Tan T, Wang Y, Quan L. Combining local features for robust nose location in 3d facial data. *Pattern Recognition Letter 2006*; **27**(13): 1487–1494.
14. D'Hose J, Colineau J, Bichon J, Dorizzi C, Dorizzi B. Precise localization of landmarks on 3d faces using gabor wavelets. In *Proceedings of BTAS '07*, 2007; 1–6.
15. Moccozet L, Dellas F, thalmann NM, *et al*. Animatable human body model reconstruction from 3d scan data using templates. In *Proceedings of CAPTECH 2004*, 2004; 73–79.
16. Mortara M, Patane G, Spagnuolo M, Falcidieno B, Rossignac J. Blowing bubbles for multi-scale analysis and decomposition of triangle meshes. *Algorithmica 2003*; **38**: 227–248.
17. Chua CS, Jarvis R. Point signatures: a new representation for 3d object recognition. *International Journal of Computer Vision (IJCV) 1997*; **250**(1): 63–85.
18. Wang Y, Chua CS, Ho. YK. Facial feature detection and face recognition from 2d and 3d images. *Pattern Recognition Letters 2002*; **23**(10): 1191–1202.
19. Deng Z, Neumann U. *Data-Driven 3D Facial Animation*. Springer-Verlag: London 2007.
20. Ersotelos N, Dong F. Building highly realistic facial modeling and animation: a survey. *The Visual Computer 2008*; **24**(1): 13–30.
21. Pighin F, Hecker J, Lischinski D, Szeliski R, Salesin D. Synthesizing realistic facial expressions from photographs. In *Proceedings of SIGGRAPH '98*, 75–84. ACM, New York, NY, USA, 1998. ACM.
22. Blanz V, Vetter T. A morphable model for the synthesis of 3d faces. In *Proceedings SIGGRAPH '99*, 1999; 187–194.
23. Joshi P, Tien WC, Desbrun M, Pighin F. Learning controls for blend shape based realistic facial animation. In *Proceedings of SCA '03*, 2003; 187–192.
24. Zhang L, Snavely N, Curless B, Seitz SM. Spacetime faces: high resolution capture for modeling and animation. In *Proceedings of SIGGRAPH '04*, 2004; 548–558.
25. Zhang Q, Liu Z, Guo B, Terzopoulos D, Shum. HY. Geometry-driven photorealistic facial expression synthesis. *IEEE Transactions on Visualization and Computer Graphics 2006*; **12**(1): 48–60.
26. Parke F, Waters K. *Computer Facial Animation*. AK Peters, Ltd.: Wellesley, Massachusetts 1996.
27. Williams L. Performance-driven facial animation. In *Proceedings of SIGGRAPH '90*, 1990; vol. 24: 235–242.
28. Guenter B, Grimm C, Wood D, Malvar H, Pighin F. Making faces. In *Proceedings of SIGGRAPH '98*, 1998; 55–66.
29. Liu Z, Shan Y, Zhang Z. Expressive expression mapping with ratio images. In *Proceedings of SIGGRAPH '01*, 2001; 271–276.
30. Chai J, Xiao J, Hodgins J. Vision-based control of 3d facial animation. In *Proceedings of SCA '03*, 2003; 193–206.
31. Vlasic D, Brand M, Pfister H, Popović J. Face transfer with multilinear models. In *Proceedings of SIGGRAPH '05*, 2005; 426–433.
32. Platt SM, Badler NI. Animating facial expressions. *SIGGRAPH Computer Graph 1981*; **15**(3): 245–252.
33. Waters K. A muscle model for animation three-dimensional facial expression. In *Proceedings of SIGGRAPH '87*, 1987; 17–24.
34. Terzopoulos D, Waters K. Physically-based facial modeling, analysis, and animation. *Journal of Visualization and Computer Animation 1990*; **1**(4): 73–80.
35. Lee Y, Terzopoulos D, Waters K. Realistic modeling for facial animation. In *Proceedings of SIGGRAPH '95*, 1995; 55–62.
36. Sumit IE, Basu S, Darrell T, Pentl A. Modeling, tracking and interactive animation of faces and heads using input from video. In *Proceedings of Computer Animation Conference'96* 1996; 68–79.
37. Zhang Y, Prakash EC, Sung E. A physically-based model with adaptive refinement for facial animation. In *Proceedings of 14th Conference on Computer Animation* 2001; 28–39.
38. Sifakis E, Neverov I, Fedkiw R. Automatic determination of facial muscle activations from sparse motion capture marker data. *Proceedings of SIGGRAPH '05*, 2005; 24(3): 417–425.
39. Capell S, Burkhart M, Curless B, Duchamp T, Popović Z. Physically based rigging for deformable characters. In *Proceedings of SCA '05*, 2005; 301–310.
40. Kähler K, Haber J, Seidel HP. Geometry-based muscle modeling for facial animation. In *Proceedings of Graphics Interface '01*, 2001; 37–46.
41. Orvalho VC, Zacur E, Susin A. Transferring the rig and animations from a character to different face models. *Computer Graphics Forum 2008*; **27**(8): 1997–2012.

Authors' biographies:

Yujian Gao is currently a PhD student in the State Key Laboratory of Virtual Reality Technology and Systems at Beihang University, China. During 2008–2009, he visited the Computer Laboratory in University of Cambridge for 1 year as visiting student. His research interests include computer facial animation, facial modelling and motion capture.

Qinping Zhao is a Professor of Computer Science in the State Key Laboratory of Virtual Reality Technology and Systems at Beihang University, China. He received his PhD degree in Computer Science from Nanjing University in 1986. His research interests include virtual reality, visualization and distribution system.

Aimin Hao is a Professor of Computer Science in the State Key Laboratory of Virtual Reality Technology and Systems at Beihang University, China. He received his PhD degree in Computer Science from Beihang University in 2006. His research interests include virtual reality, photorealistic rendering and GPU based acceleration.

T.M. Sezgin is an Assistant Professor in the College of Engineering at Koç University in Istanbul and leads the Intelligent User Interfaces Laboratory. He graduated from Syracuse University in 1999, then completed his MS in the CSAIL at MIT in 2001. He received his PhD in 2006 from MIT, and subsequently joined the Rainbow group at the University of Cambridge Computer Laboratory as a Postdoctoral Research Associate. His research interests include intelligent human–computer interfaces, multimodal sensor fusion, and HCI applications of machine learning.

Neil Dodgson is Reader in Graphics and Imaging in the Computer Laboratory at the University of Cambridge, where he is a co-leader of the Graphics and Interaction Research Group (Rainbow). He graduated from Massey University in 1987, then received his PhD in the Computer Laboratory at University of Cambridge in 1992. Dr Dodgson's research interests are in computer graphics, 3D display technology and image processing. His recent research has focussed on subdivision surfaces and aesthetic imaging.

DOI: 10.1002/cav