COMPUTER ANIMATION AND VIRTUAL WORLDS
Comp. Anim. Virtual Worlds 2012; **23**:167–178
Published online 30 May 2012 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cav.1455

SPECIAL ISSUE PAPER

Video-driven state-aware facial animation

Ming Zeng[1], Lin Liang[2], Xinguo Liu[1]* and Hujun Bao[1]

[1] State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang, China
[2] Microsoft Corporation, Seattle, WA, USA

ABSTRACT

It is important in computer animation to synthesize expressive facial animation for avatars from videos. Some traditional methods track a set of semantic feature points on the face to drive the avatar. However, these methods usually suffer from inaccurate detection and sparseness of the feature points and fail to obtain high-level understanding of facial expressions, leading to less expressive and even wrong expressions on the avatar. In this paper, we propose a state-aware synthesis framework. Instead of simply fitting 3D face to the 2D feature points, we use expression states obtained by a set of low-cost classifiers (based on local binary pattern and support vector machine) on the face texture to guide the face fitting procedure. Our experimental results show that the proposed hybrid framework enjoys the advantages of the original methods based on feature point and the awareness of the expression states of the classifiers and thus vivifies and enriches the face expressions of the avatar. Copyright © 2012 John Wiley & Sons, Ltd.

KEYWORDS
facial animation; face state recognition; video-driven animation; state-aware face action estimation

***Correspondence**
Xinguo Liu, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang, China.
E-mail: xgliu@cad.zju.edu.cn

1. INTRODUCTION

Driving an avatar face from a video without any marker has a wide range of applications in computer animation. To reflect the facial expressions and emotions on the avatar, many researchers have proposed various approaches over the past decades. One category of most frequently used methods is based upon 2D feature points [1–3]. These methods first detect some semantic points on the face (e.g., eye/mouth corners, etc.) and then drive the avatar according to the movements of these points. However, these methods largely suffer from the inaccurate locations and the sparseness of the detected points (Figure 1), which results in less expressive or wrong emotions on the avatar face.

Complementary to feature points, the texture of the face image contains abundant expression information and is more stable. On the basis of this observation, we propose a hybrid real-time video-based face synthesis system that uses feature points to estimate movements of the face and employs LBP+SVM classifiers to recognize expression states from the face texture to avoid wrong expressions. Enjoying the advantages of the feature points in continuous position estimation (though inaccurate) and the expression states recognition from classifiers, the avatar synthesized by our system accurately reflects the emotions of human and is able to make various vivid expressions. Experiments and a user study show that our system generates much better animation than the method based only on feature points.

This paper presents a novel synthesis framework for video-based facial animation, which contains the following technical contributions:

- Firstly, we propose an accurate and efficient expression state recognition method that utilizes both state transition priors and an online template technique to improve recognition rate and computing efficiency.
- Secondly, we design a state-aware face action estimation method to fuse information of feature points and face texture, which makes our system be aware of both low-level features and high-level expressions on the face.

2. RELATED WORK

Performance-driven facial animation (PDFA) is a class of techniques to animate faces according to human's performance. Inputs of PDFA include image/video, motion capture data, depth data [4,5], and so on. Here, we focus on video-based facial animation. For a global survey of facial animation and PDFA, readers can refer to [6–8].

Copyright © 2012 John Wiley & Sons, Ltd.

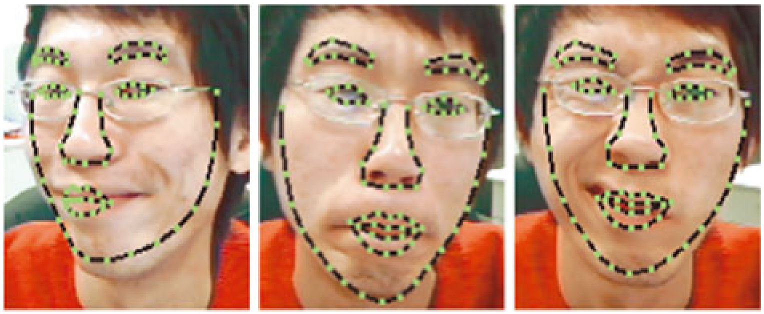

Figure 1. Feature points: (a) cannot capture mouth corners on sideview; (b) cannot fit the closed mouth well and cannot capture texture changes between two brows; (c) fail to fit some special expression.

2.1. Video-based Facial Animation

There are mainly three kinds of methods for video-based facial animation: optical flow based, feature point based, and appearance based.

Methods based on optical flow [9–11] used optical flow to build correspondences in successive frames of videos (images) to drive the 3D face model. However, tracking methods based on optical flow usually suffer from accumulation of errors, known as drift.

Methods based on feature point [1–3] automatically locate and track some 2D feature points to animate 3D face models. Feature points are flexible to capture deformable shapes but not stable and accurate enough.

Appearance-based methods [12–15] iteratively synthesize face to approximate the texture of current face and estimate the face motion. These methods are stable because of their patch-based distance metrics. However, they cannot capture detailed changes well.

Our system is built upon a face tracker based on active appearance model [16]. Although the tracker is based on appearance, we do not directly use the parameters. Instead, we adopt semantic landmarks as feature points to drive avatars. In this point, our method should be regarded as feature point based.

2.2. Facial Expression Recognition

In our system, we use facial expression recognition to determine the states of the face expressions. Facial expression recognition [17–19] classifies a face image into seven kinds of prototypic expressions (neutral, happiness, surprise, anger, disgust, sadness, and fear) [20] or a set of facial actions units (AU) [21].

2.3. Expressive Avatar Synthesis

To improve the expressiveness of synthesized avatars, many researchers have made great efforts. Zalewski and Gong [22] analyzed face image via expression recognition and used the recognition confidence as the coefficients of blendshapes. However, Zalewski's method is person specific, and in fact, the recognition confidence is not equal to the magnitudes of facial movements. Deng and his colleagues developed eFace [23] for expressive facial animation synthesis and editing based on mocap data and introduced a facial perceptual metric FacePEM [24]. These methods produced vivid expressions but required mocap database or user perceptual scores as the training data. More recently, Taehyum Rhee *et al.* [25] proposed a video-based facial animation pipeline. They used edge information to correct the feature points but did not capture high-level face expressions, which will inevitably lead to wrong expressions. In our work, we take the advantages of facial expression recognition and face feature points tracking to generate vivid and expression-correct facial animations. Our motivation is similar to the work of Fidaleo *et al.* [26]. Fidaleo *et al.* separately used 2D features to morph shape and used classifiers to transfer texture animation, whereas our method not only uses classifiers to capture the changes of winkle and crease but also uses them to guide the fitting procedure, which synthesizes facial animations with correct expressions.

3. SYSTEM OVERVIEW

The input of our system is a facial performance video, and the output is an animated avatar with corresponding facial expressions. In our system, users can make expressions in front of a web camera to directly control the 3D avatar head in real time. As is illustrated in Figure 2, our system consists of four main components:

Feature points tracker uses a face tracker based on active appearance mode [16] to track feature points (see Figure 5(a)) on the face.

Face state classifiers adopt a set of classifiers to recognize expression states of regions of the face (called *face states*). These regions are located and cropped on the basis of the feature points extracted from the previous stage.

State-aware face action analyzer estimates the coefficients of a set of bases of a morphable face model [27] according to the feature points and the face states.

Comp. Anim. Virtual Worlds 2012; **23**:167–178 © 2012 John Wiley & Sons, Ltd.
DOI: 10.1002/cav

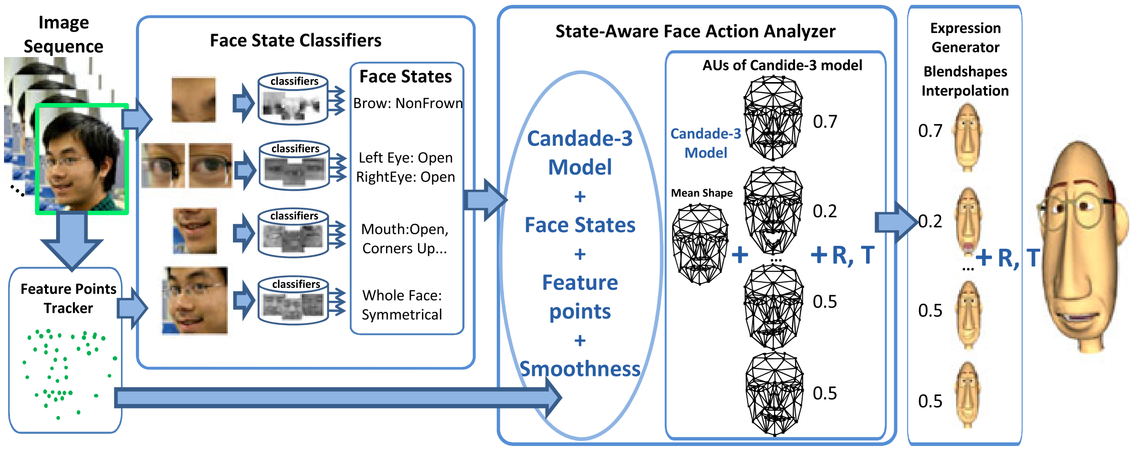

Figure 2. The overview of our system.

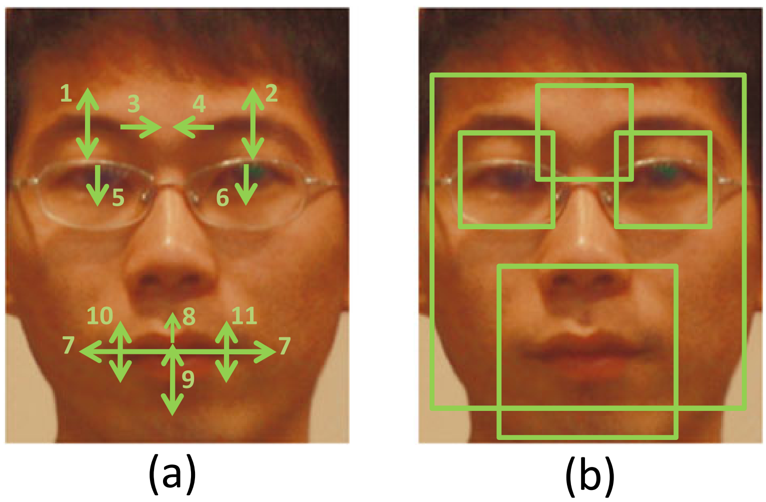

Figure 3. (a) Face actions; (b) texture regions.

Expression generator uses the coefficients obtained in the state-aware face action analyzer to synthesize the expressions on the avatar based on blendshape interpolation [1].

4. FACE STATE RECOGNITION

Facial expressions are formed by a set of facial movements. In our system, we define 11 movements, including four brow movements, two eye movements, and five mouth movements (see Figure 3(a)). These movements lead to texture changes on some regions of the face (rectangles in Figure 3(b)). For each region in Figure 3(b), we use classifiers on its texture to recognize their face states. We define the following face states:

Brow frown states $F = \{0, 1\}$ represents whether the brows frown or not.

Eye closure states $E_i = \{0, 1\}$, where $i \in \{\text{left}, \text{right}\}$, represents whether an eye is closed or not.

Mouth emotion states $M_{\text{emo}} = \{-1, 0, 1\}$ represents whether the mouth is unhappy (mouth corners down), neutral (mouth corners keep neutral in vertical direction), or happy (mouth corners up).

Mouth shape states $M_{\text{shp}} = \{-1, 0, 1\}$ represents whether the mouth shape is pouty (mouth corners inner), normal (mouth corners keep neutral in horizonal direction), or stretched (mouth corners outer).

Mouth closure states $M_{\text{clo}} = \{0, 1\}$ represents whether the mouth is open or closed.

Symmetrical states $S = \{-1, 0, 1\}$ represents whether the whole face is left askew (left mouth corner up, right mouth corner down, left eye closed, right eye opened), symmetrical, or right askew (contrast to left askew).

These six kinds of face states describe the high-level understanding of one face image.

4.1. Face State Classifier

We extract the local binary pattern (LBP) features [18] from the corresponding face regions and recognize the face states using support vector machine (SVM) [28]. For a face state set with n state values, we adopt the one-versus-all scheme [29] to train n binary classifiers.

4.2. States Probabilistic Analysis

4.2.1. Probabilistic Representation

We formulate the classification problem in a probabilistic form. For each binary classifier, we denote the distribution function of the positive samples by F_{p} and the negative samples by F_{n}. The probability of a sample belonging to the positive class is defined as

$$P(x) = F_{\mathrm{p}}(x) / \left(F_{\mathrm{p}}(x) + F_{\mathrm{n}}(x)\right), \tag{1}$$

where F_{p} and F_{n} can be approximated by the discrete normalized histogram of the positive and the negative samples, respectively, which can be estimated from the training data. Then we use the following sigmoid function to fit $P(x)$:

$$1/(1 + exp(-\phi x + \psi)) \tag{2}$$

by the gradient descent method. After solving the parameters ϕ and ψ in Equation 2, we transform the confidence score x to probability. The probability of each class is further normalized by dividing their sum.

4.2.2. Maximum a Posteriori Estimation.

To avoid animation dithering due to recognition error, we introduce expression state transition probability as the prior knowledge. With the prior, the expression recognition problem can be formulated:

$$s^{\star} = argMax_{s_i} P\left(S_t = s_i | \widehat{S}_{t-1}, I_t\right), \tag{3}$$

where $\widehat{S}_{t-1}$ denotes the observed state at $t-1$ and I_t denotes the image patch. Assuming uniform distribution of S_t and independence between $\widehat{S}_{t-1}$ and I_t, the equation can be further expanded as:

$$s^{\star} = argMax_{s_i} P\left(S_t = s_i | \widehat{S}_{t-1}\right) \cdot P\left(S_t = s_i | I_t\right), \tag{4}$$

where $P\left(S_t = s_i | \widehat{S}_{t-1}\right)$ is the state transition probability conditioned on the observed state $\widehat{S}_{t-1}$. The probability can be set empirically or estimated from the training data. In our implementation, we set high probability $P_h = 0.6$ for keeping the observed state and averagely distribute the remaining probability $1 - P_h$ among the other states. $P(S_t = s_i | I_t)$ is the probability of face state s_i conditioned on the cropped patch I_t, which is given by the normalized results of Equation 2 for the corresponding classifier.

4.3. Online Template Technique

We propose an online template technique to further improve the recognition rate and save computation time. We store some patches that have very high recognition scores at runtime and call them template patches. When a new image patch comes, we first compare it with the templates. If they are similar enough, we directly use the state of the matched template as that of the current patch. Otherwise, we use the SVM classifiers and the maximum a posteriori estimation to classify the patch. If the score of the patch belonging to one state is high enough, we consider the patch as a new template and update the template set. For a patch $\mathcal{P}$ and a template $\mathcal{Q}$, we use $L1$ distance $|\mathcal{P} - \mathcal{Q}|$ between their intensities to efficiently compute the similarity.

5. FACE ACTION ANALYZER

In this part, we use feature points and face states to determine the magnitudes of face actions on the basis of a 3D morphable face model.

5.1. 3D Face Morphable Model

We use Candide-3 [27] as our morphable model, which is a parameterized wireframe face model, defined by a mean shape $\overline{g}$, a set of shape units S and a set of face action units A. The shape unit set S gives the basic shapes (head height, eyes distance, etc.) of human faces, and the face action unit set A represents the actions of the face. An expression can be represented as:

$$g(\sigma, \alpha) = \overline{g} + S\sigma + A\alpha, \tag{5}$$

where σ and α are coefficients controlling shape units and action units, respectively.

In our implementation, we use 11 AUs as listed in Table I. We show AUs in Figure 4 and give the ranges of coefficients of each AUs and their descriptions in Table I.

5.2. Direct Fitting

To estimate the face expression from an image, a straightforward method is to find a rigid transformation (R, T) and the action units coefficients α in Equation 5 to determine a face geometry that best fits the extracted 2D feature points (usually, σ is determined at initialization) in following measurement:

$$E_{\mathrm{fit}} = \sum_{i \in V} \left\| \Pi\left(R\left(g^i\right) + T\right) - P_{2d}^i \right\|^2, \tag{6}$$

where $\Pi : \mathcal{R}^3 \rightarrow \mathcal{R}^2$ is a perspective projection of the pinhole camera model. P_{2d} is the locations of the 2D feature points. V is the points set used as the fitting

 Comp. Anim. Virtual Worlds 2012; **23**:167–178 © 2012 John Wiley & Sons, Ltd.
DOI: 10.1002/cav

Table I. Descriptions of actions units (AUs).

AU(ID)	Ranges of Coefficients and Descriptions
LBV(1)	(0, 1], Left brow raiser
	[-1, 0], Left brow lower
RBV(2)	(0, 1], Right brow raiser
	[-1, 0], Right brow lower
LBI(3)	[0, 1], Left brow frown
RBI(4)	[0, 1], Right brow frown
LEC(5)	[0, 1], Left eye closed
REC(6)	[0, 1], Right eye closed
DCH(7)	(0, 1], Mouth stretcher
	[-1, 0], Mouth pucker (pouty)
CLV(8)	[0, 1], Mouth upper lip open
CJV(9)	[0, 1], Mouth jaw drop
LCV(10)	(0, 1], Left mouth corner puller
	[-1, 0], Left mouth corner depressor
RCV(11)	(0, 1], Right mouth corner puller
	[-1, 0], Right mouth corner depressor

constraints (Figure 5). However, the extracted 2D feature points may not be accurate enough, which leads to wrong face expressions.

5.3. State-aware Face Action Estimation

We propose a state-aware face action estimation scheme to avoid generating wrong expressions. As is shown in Algorithm 1, our method consists of the following steps:

(1) **Initialize coefficients of shape units.** On the first frame, we assume all the coefficients of action units are zeros and estimate coefficients of shape units by minimizing Equation 6. In successive frames, we keep the estimated σ as constants and only estimate R, T, and α.

(2a) **Pose estimation.** We minimize Equation 6 to estimate the face pose R and T. Here, we constrain the expression to be symmetrical to improve the robustness of pose estimation. Therefore, the coefficients of the corresponding AUs on the two sides of the face are equal: $\alpha_{\mathrm{LBV}} = \alpha_{\mathrm{RBV}}$ and $\alpha_{\mathrm{LCV}} = \alpha_{\mathrm{RCV}}$. Considering the qualitative estimation of action unit coefficients α in the following steps, we do not adopt an iterative manner to re-estimate R and T according to α.

(2b) **States-aware face action fitting.** In this step, we fix R and T estimated from the previous pose estimation and only estimate the coefficients of AUs. We regard the face expression as symmetrical for now and deal with the asymmetrical expressions in the step of asymmetrical modification.

State-aware constraint. After the face state classifiers judging the states of the face, we expect the coefficients of the corresponding action units α to be consistent with the estimated states. We assume that the coefficients controlling mouth shape (α_{DCH}) and mouth emotion ($\alpha_{\mathrm{LCV}}, \alpha_{\mathrm{RCV}}$) are Gaussian-distributed variables conditioned on the estimated

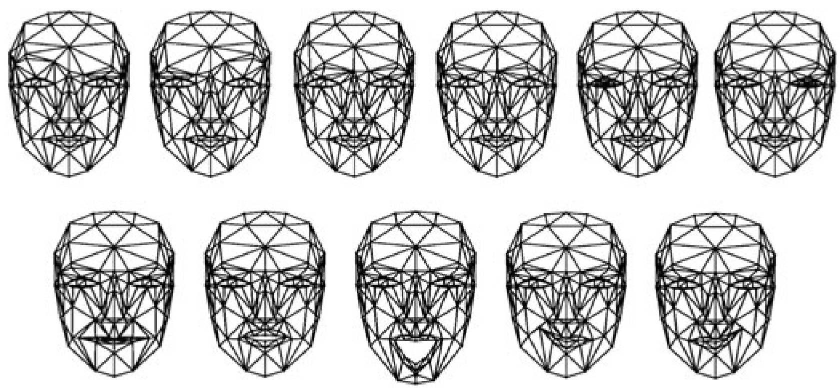

Figure 4. Face action units. The first row: LBV, RBV, LBI, RBI, LEC, REC. The second row: DCH, CLV, CJV, LCV, RCV.

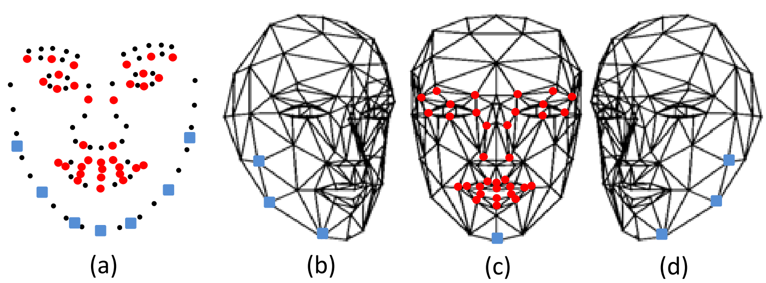

Figure 5. 2D–3D feature points correspondence.

DOI: 10.1002/cav

face states. Therefore, we use the negative logarithm of their distribution functions to constrain the movements of the corresponding parts to be not far away from the recognized states:

$$E_{\mathrm{shp}} = \frac{\left(\alpha_{\mathrm{DCH}} - e_{\mathrm{shp}}\right)^2}{\sigma_{\mathrm{shp}}^2}, \tag{7}$$

$$E_{\mathrm{emo}} = \sum_{\alpha_i \in \{\alpha_{\mathrm{LCV}}, \alpha_{\mathrm{RCV}}\}} \frac{\left(\alpha_i - e_{\mathrm{emo}}\right)^2}{\sigma_{\mathrm{emo}}^2}. \tag{8}$$

E_{shp} constrains the movement of mouth corners in the horizontal direction (AU7) to form the recognized mouth shape state (Figure 6(a)). E_{emo} constrains left/right mouth corners (AU10/AU11) to form the recognized mouth emotion state (Figure 6(b)). The means and variances of the Gaussian distributions are listed in Table II.

To make the reconstructed mouth be closed, we introduce a mouth-close term

$$E_{\mathrm{clo}} = \begin{cases} \rho \left\| P_1(g) - P_2(g) \right\|^2, & M_{\mathrm{clo}} = 1 \\ 0, & M_{\mathrm{clo}} = 0 \end{cases} \tag{9}$$

where $P_1(g)$ and $P_2(g)$ are the positions of two points on the inner contour of the mouth (red dots in Figure 6(c)). E_{clo} penalizes separation of the upper and lower lips when the mouth is recognized as closed. The parameters ρ is set to 100 in our implementation.

Combining Equations 7, 8, and 9, we have the following state energy as the state constraints:

$$E_{\mathrm{sta}} = E_{\mathrm{shp}} + E_{\mathrm{emo}} + E_{\mathrm{clo}}. \tag{10}$$

Smoothness constraint To smooth the change speed of the coefficients of the action units, we define the following smoothness term:

$$E_{\mathrm{smo}} = \sum_j \left\| \alpha_t^j - \alpha_{t-1}^j \right\|^2. \tag{11}$$

Combining Equations 6, 10, and 11, we obtain the following extended energy function for state-aware face action fitting:

$$E = \frac{1}{d} E_{\mathrm{fit}} + w_{\mathrm{sta}} E_{\mathrm{sta}} + w_{\mathrm{smo}} E_{\mathrm{smo}}, \tag{12}$$

where d is the length of the diagonal line of the 2D face bounding box in pixels. In our current implementation, we use the weights $\omega_{\mathrm{sta}} = 0.1$ and $\omega_{\mathrm{smo}} = 2$. We adopt the method of Gauss–Newton to figure out the optimal α.

(2c) **Upper face and asymmetrical modification.** Because our face state classifiers can analyze the texture changes caused by brow frown and eye closure, which are hard to be estimated by the feature points, we directly set the coefficients of left brow frown (AU 3), right brow frown (AU 4), left eye close (AU 5), and right eye close (AU 6) to presence or absence (1.0 or 0.0) according to face states.

For the asymmetrical expressions, we also adjust the coefficients according to the face states. If the face is left askew, then modify related AUs (left eye close (AU 5), right eye open (AU 6), left mouth corner up (AU 10), right mouth corner down (AU 11)). If the face is right askew, then make opposite modifications.

Algorithm 1 State-aware face action estimation.

1. Initialize coefficients of shape units
2. Face action estimation

(a) Pose estimation

(b) State-aware face action fitting

(c) Upper face and asymmetrical modification

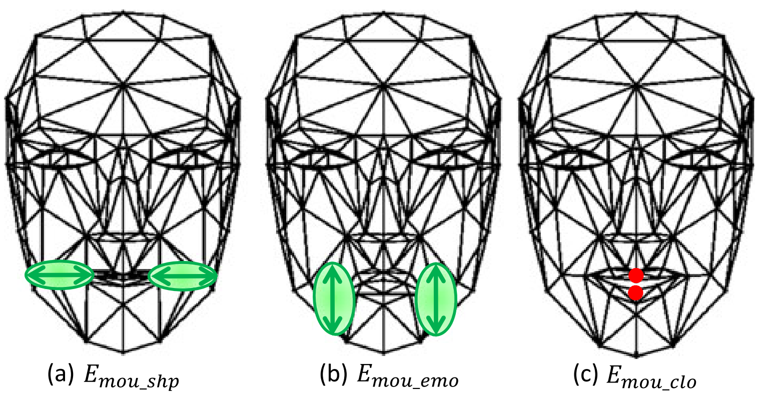

Figure 6. State-Aware Constraints.

Comp. Anim. Virtual Worlds 2012; **23**:167–178 © 2012 John Wiley & Sons, Ltd.
DOI: 10.1002/cav

6. EXPRESSION GENERATOR

We transfer coefficients of face action units to a set of blendshapes of an avatar to generate expression animations. As shown in Figure 7, there are 17 blendshapes, which are designed according to the AUs. We directly set the coefficient β of these blendshapes as the coefficients α of corresponding AUs (note that in the blendshapes, all coefficients are in $[0, 1]$, so the negative coefficients of the AUs correspond to the blendshapes with opposite direction actions).

At the end of this stage, we use previous four frames to smooth the current result (except eye parts) in a linear weighting scheme, and the weight is $w = \cos(0.5\beta\pi)$. Over the transition duration, β moves linearly from 1 (earliest) to 0 (current).

Table II. Parameters of distribution of α conditioned on mouth states.

Mouth shape	Stretcher	Normal	Pouty
(e, σ^2)	(1.0, 0.01)	(0.0, 0.01)	(−1.0, 0.01)
Mouth emotion	Happy	Neutral	Unhappy
(e, σ^2)	(1.0, 0.01)	(0.0, 0.01)	(−1.0, 0.01)

7. EXPERIMENTS

7.1. Face State Recognition

7.1.1. Feature Extraction and Classifier.

For each classifier, we locate and crop the corresponding patches (see Figure 3(b)) according to the feature points. We resize the whole face patch to 44×44 pixels, and other patches to 30×30 pixels. All image patches are gray scaled and histogram equalized.

We extract the LBP [18] from the image patches. For each patch, we left one pixel on four sides of its boundary and evenly distribute 25 and 36 equally sized grid blocks on the 30×30 and 44×44 patches, respectively. For each grid blocks, a 59-bins LBP histogram of the grid encoding the texture variance is counted. Histograms of all grid blocks are then concatenated to form LBP feature vector of this image patch. After the feature extraction, we use linear SVM [28] to obtain the state score.

7.1.2. Training Dataset.

To train the SVM classifiers, we build up a dataset using various facial expressions from public datasets: CMUPIE [30], CMUAU [31], LFW [32], and our own dataset. The final dataset (Figure 8) contains thousands of people with different skin colors, different ages, different genders, frontal view, and side view. For each face state, we extract the patches (Figure 3(b)) and manually label them as pos/neg samples. The statistics of the training

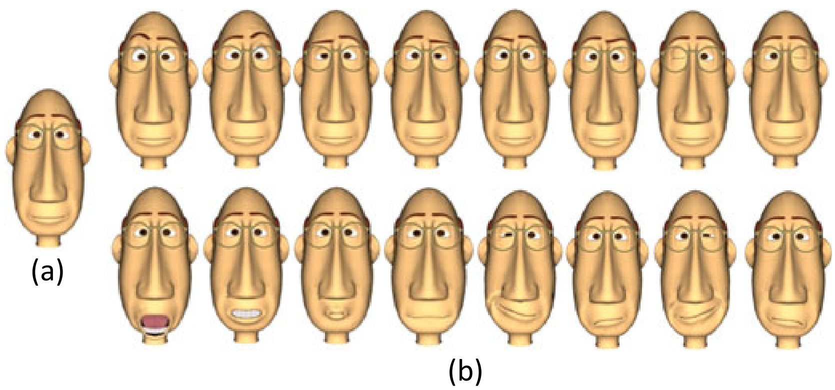

Figure 7. (a) Neutral face; (b) 16 blendshapes.

Figure 8. Training Datasets.

Table III. Statistics of classifiers.

Classifier	# train samples	
Brow (Fr/NFr)	1089/4995	
Eye (Clo/Ope)	15,024/29,664	
Mou (Clo/Ope)	45,694/27,757	
Mou (Hap/Neu/NHap)	9126/30,060/14,782	
Mou (Pty/Nor/Str)	10,676/47,518/14,938	
Asym (Lef/Nor/Rig)	13,872/26,220 /13,872	
Classifier	**# test samples**	**Accu**
Brow (Fr/NFr)	468/2151	95.69%
Eye (Clo/Ope)	10,128/19,872	97.41%
Mou (Clo/Ope)	27,000/18,648	95.06%
Mou (Hap/Neu/NHap)	6102/20,061/5196	94.43%
Mou (Pty/Nor/Str)	3627/35,919/5400	96.33%
Asym (Lef/Nor/Rig)	4692/16,752/4692	94.48%

and testing data for all classifiers are listed in Table III. This table shows the reliability of recognition using basic SVM classifiers.

7.1.3. Recognition Performance.

We compare expression recognition results on two sequences using C_1 (original classifiers), C_2 (C_1+expression prior), and C_3 (C_2+online template), respectively. In Figures 9 and 10, we show recognition results on two sequences "eyeState" and "asymFace". In these figures, the first row is the ground truth, second, third, and fourth rows are the results of C_1, C_2, and C_3, respectively, and the bottom row is hit/miss statistics of the online templates matching. In the two comparisons, our classifiers outperform original classifiers on recognition accuracy and produce much more smooth results, although a little dragging effect (97th–98th frame of "asymFace").

7.2. Comparisons with Direct Fitting

Our state-aware method is able to synthesize much more expressive avatar than direct fitting method. We show their comparisons in Figure 11. The top 2 rows show that our state-aware fitting procedure can guide the fitting by adding state-aware terms, so as to produce correct expressions (smile on side view and pouty closed mouth). In the third row, although our state-aware fitting procedure fails to correct the fitting points (d), our method qualitatively adjusts the coefficients for related action units in the last step of our state-aware estimation method (f), which leads to a correct asymmetrical expression (h).

We also conduct a user study to compare our method with direct fitting. We synthesize animations of an avatar by our method and direct fitting, respectively, according to a human expression video. Ten volunteers are asked to tell which animation is better and to pick top five frames that support their opinions most. They are also required to score the two animations in four facets: expressiveness, nature, smoothness, and accuracy from 1 (worst) to 5 (best).

All volunteers consider that our result is better than direct fitting's. We show some snapshots chosen by participators to support our method in Figure 12 and the scores of the user study in Figure 13. It is clear in Figure 13 that our method is better than direct fitting in facets of expressiveness, nature, and accuracy, and both of them get same scores in smoothness.

7.3. Expressiveness and Generality

Figure 14 shows the ability of our method to generate vivid expressions. Besides, we show synthesis results for various avatar faces in Figure 15 according to one face sequence (first row). The synthesis results include a photorealistic human face (second row), a human-like cartoon face (third row), and a box-shaped face (fourth row).

7.4. Computing Efficiency

On a machine with Pentium 4 3.0 GHz CPU and 1 GB RAM, our system (from sequence to rendered results) can run at about 40 fps with 320×240 resolution. A great deal of (more than 60%) computation of classifiers is avoided by the online template technique (e.g., H/M rows of

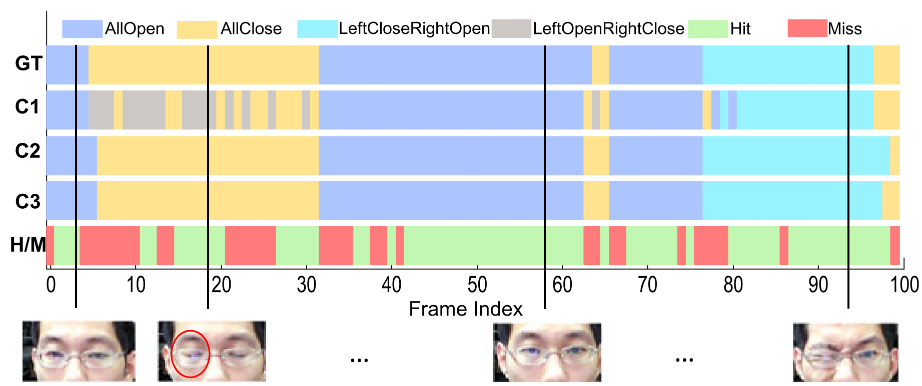

Figure 9. Comparison results on "eyeState". C_1 produces jitters in frames 5 ∼ 30 because of reflection on eye glasses. C_2 and C_3 filter jitters.

Comp. Anim. Virtual Worlds 2012; **23**:167–178 © 2012 John Wiley & Sons, Ltd.
DOI: 10.1002/cav

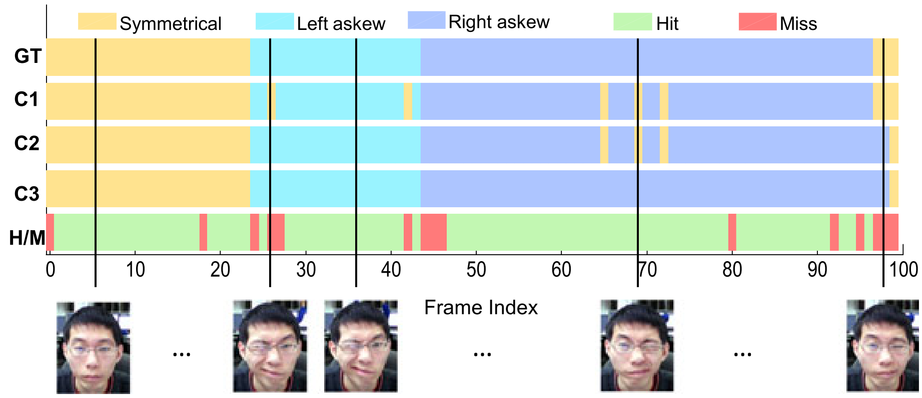

Figure 10. Comparison results on "asymFace". C_1 wrongly recognizes the 69*th* frame, and C_2 fails to correct it. C_3 successfully matches it with one template and recognizes the right state.

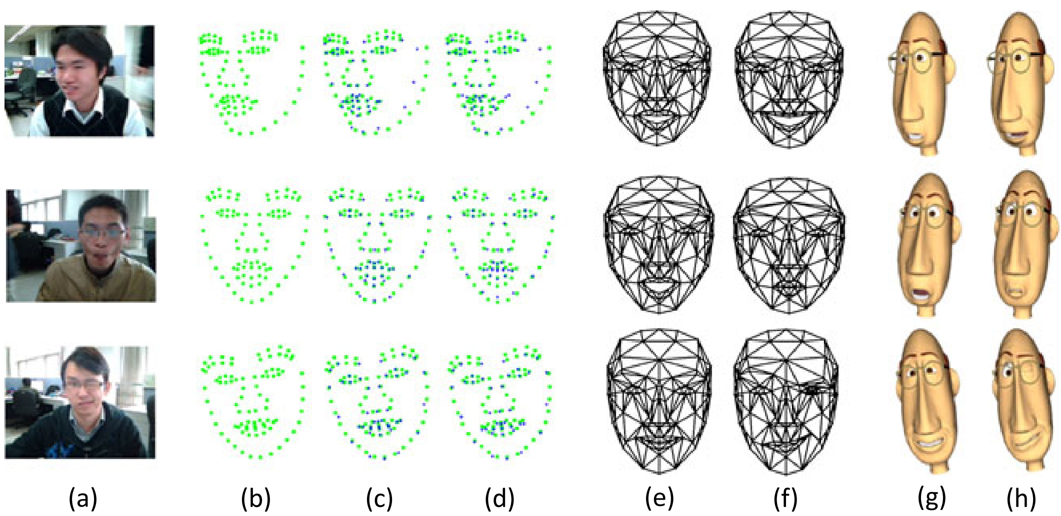

Figure 11. Comparisons with direct fitting. (a) Input image, (b) 2D feature points, (c) direct fitting results, (d) state-aware fitting results, (e) estimated mesh by direct fitting, (f) estimated mesh by state-aware estimation, (g) synthesis avatar by direct fitting, and (h) synthesis avatar by state-aware estimation.

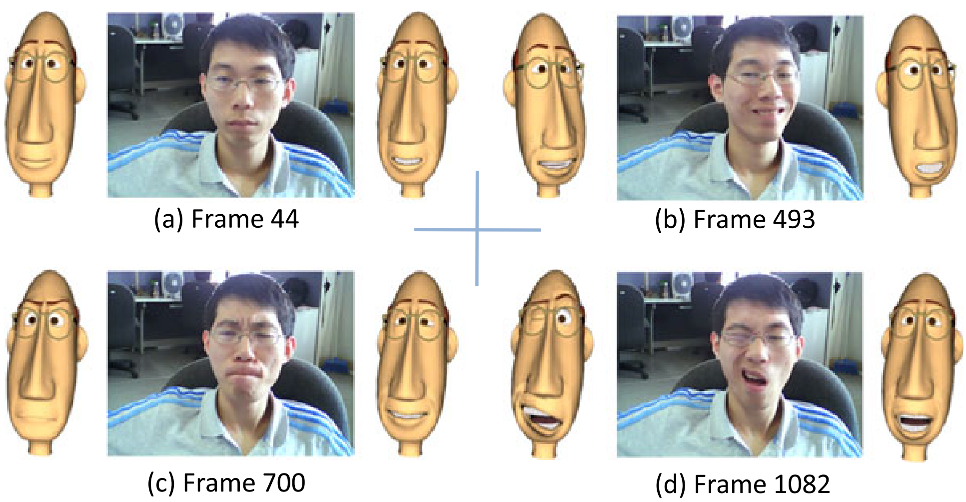

Figure 12. Comparison examples. For each example, the left one is our result, and the right one is direct fitting's.

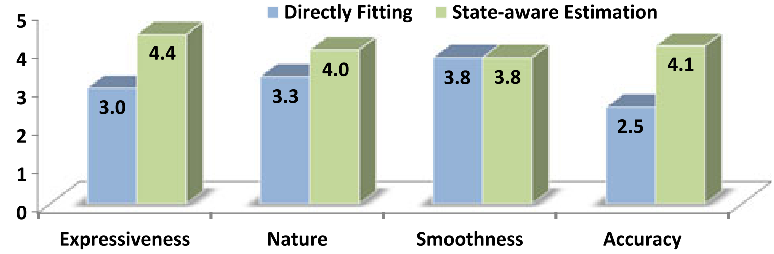

Figure 13. Scores of our user study.

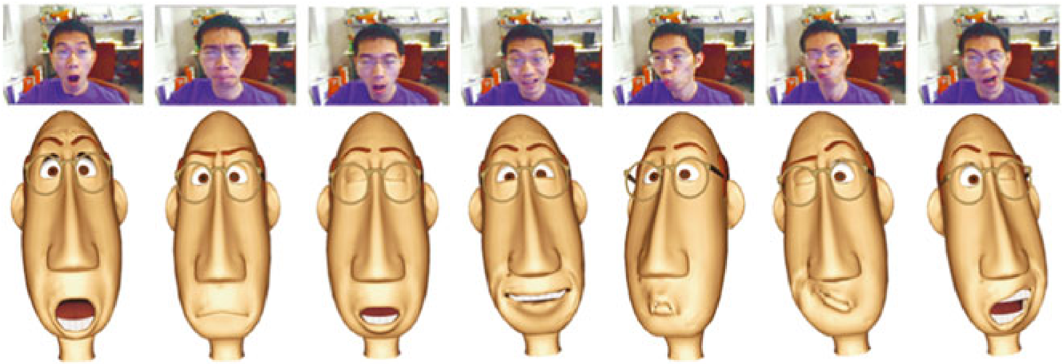

Figure 14. Expression examples. From left to right: surprise, upset, sleepiness, surprise+happyness, pouty mouth, pouty+asym face, and open mouth+asym face.

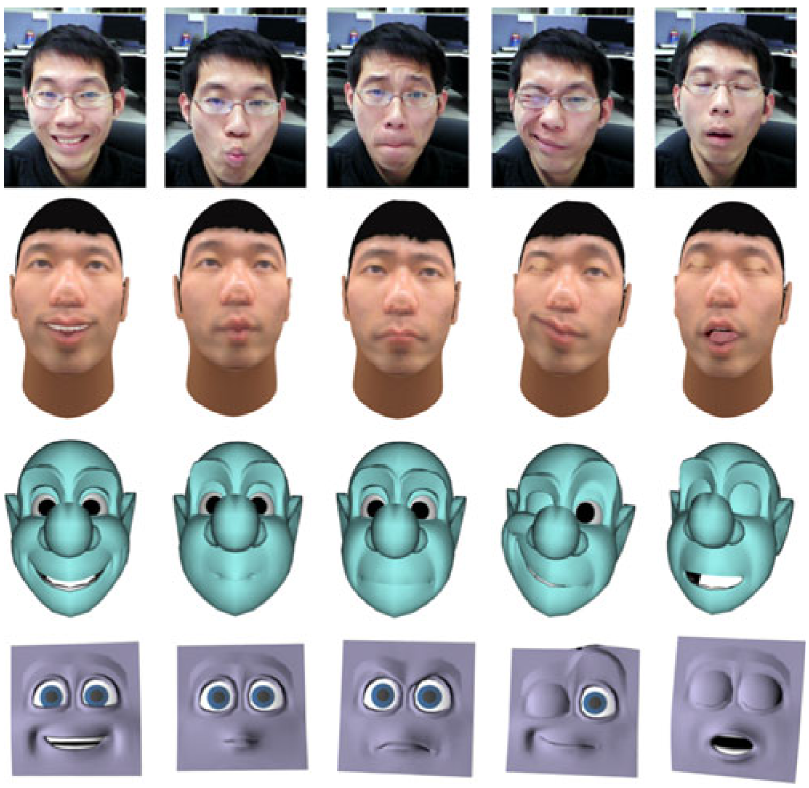

Figure 15. More results. The first row is the input sequence, and the second to fourth rows are synthesis results on difference faces.

Comp. Anim. Virtual Worlds 2012; **23**:167–178 © 2012 John Wiley & Sons, Ltd.
DOI: 10.1002/cav

Figures 9 and 10), which largely improves the efficiency of our system.

8. CONCLUSIONS AND FUTURE WORK

We propose a novel real-time state-aware video-driven face animation method. We utilize classifiers to capture the high-level understanding of the face states and then guide the procedure of the face action estimation, which leads to much more vivid and expression-correct facial animations. The experiments with a user study demonstrate the expressiveness of our results and the generality and efficiency of our system.

Our approach works well, but there is still space for improvement. Firstly, the current system cannot handle side views larger than 40° well. This is because the training data for our face state classifiers does not contain these examples, which can be addressed by adding more training data for different views. Besides, more special expressions can be supported by training more expression classifiers.

ACKNOWLEDGEMENTS

Many thanks to Xin Tong and Jian Sun for fruitful discussions; Luoying Liu, Tao Wang, and Lin Zhou for providing the CG characters; and the anonymous reviewers for their valuable comments. This work was partially supported by NSFC (No. 60970074), China 973 Program (No. 2009CB320801), Fok Ying Tung Education Foundation, and the Fundamental Research Funds for the Central Universities.

REFERENCES

1. Chuang E, Bregler C. Performance driven facial animation using blendshape interpolation. *Technical Report SU-CS-TR-2002-02*, Stanford University, 2002.
2. Chai J, Xiao J, Hodgins J. Vision-based control of 3D facial animation. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2003; 193–206.
3. Zhang W, Wang Q, Tang X. Real time feature based 3-D deformable face tracking. In *Proceedings of the 10th European Conference on Computer Vision: Part II*, ECCV '08. Springer-Verlag, Berlin, Heidelberg, 2008; 720–732.
4. Cai Q, Gallup D, Zhang C, Zhang Z. 3D deformable face tracking with a commodity depth camera. In *Proceedings of the 11th European Conference on Computer Vision Conference on Computer Vision: Part III*, ECCV'10. Springer-Verlag, Berlin, Heidelberg, 2010; 229–242.
5. Weise T, Bouaziz S, Li H, Pauly M. Realtime performance-based facial animation. *ACM Transactions on Graphics* 2011; **30**(4): 77:1–77:10.
6. Williams L. Performance-driven facial animation. *Computer Graphics (SIGGRAPH '90 Proceedings)* 1990; **24**(4): 235–242.
7. Parke W. *Computer Facial Animation*. A.K.Peters, Wellesley, MA, 1996.
8. Williams L. Performance-driven facial animation. In *Acm SIGGRAPH 2006 Courses*, SIGGRAPH '06. ACM, New York, NY, USA, 2006.
9. Blanz V, Vetter T. A morphable model for the synthesis of 3D faces. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 1999; 187–194.
10. Decarlo D, Metaxas D. Optical flow constraints on deformable models with applications to face tracking. *International Journal of Computer Vision* 2000; **38**(2): 99–127.
11. Vlasic D, Brand M, Pfister H, Popović J. Face transfer with multilinear models. *ACM Transactions on Graphics* 2005; **24**(3): 426–433.
12. Li H, Roivainen P, Forcheimer R. 3-D motion estimation in model-based facial image coding. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1993; **15**(6): 545–555.
13. Pighin F, Szeliski R, Salesin DH. Resynthesizing facial animation through 3d model-based tracking, In *ICCV*, Kerkyra(Corfu), Greece, 1999; 143–150.
14. Pandzic I, Forchheimer R. *MPEG-4 Facial Animation: The Standard, Implementation and Application*. Wiley, New York, USA, 2002.
15. Chuang E, Bregler C. Mood swings: expressive speech animation. *ACM Transactions on Graphics* 2005; **24**(2): 331–347.
16. Zhou M, Liang L, Sun J, Wang Y. AAM based face tracking with temporal matching and face segmentation. In *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, San Francisco, CA, USA, 2010; 701–708.
17. Littlewort G, Bartlett MS, Fasel I, Susskind J, Movellan J. Dynamics of facial expression extracted automatically from video. *Image and Vision Computing* 2006; **24**(6): 615–625.
18. Shan C, Gong S, McOwan PW. Facial expression recognition based on local binary patterns: a comprehensive study. *Image and Vision Computing* 2009; **27**(6): 803–816.
19. Zeng Z, Pantic M, Roisman GI, Huang TS. A survey of affect recognition methods: audio, visual,

and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2009; **31**(1): 39–58.
20. Ekman P, Davidson RJ. *The Nature of Emotionl Fundamental Questions*. Oxford University Press, New York, 1994.
21. Ekman P, Friesen WV. *Facial Action Coding System (FACS): Manual*. Consulting Psychologists Press, Palo Alto, USA, 1978.
22. Zalewski L, Gong S. 2D statistical models of facial expressions for realistic 3D avatar animation. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)—Volume 2*, CVPR '05. IEEE Computer Society, Washington, DC, USA, 2005; 217–222.
23. Deng Z, Neumann U. efase: expressive facial animation synthesis and editing with phoneme-isomap controls. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '06. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2006; 251–260.
24. Deng Z, Ma X. Perceptually guided expressive facial animation. In *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '08. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2008; 67–76.
25. Rhee T, Hwang Y, Kim JD, Kim C. Real-time facial animation from live video tracking. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '11. ACM, New York, NY, USA, 2011; 215–224.
26. Fidaleo D, Noh J, Kim T, Enciso R, Neumann U. Classification and volume morphing for performance-driven facial animation, In *International Conference on Digital and Computational Video*, Tampa, USA, 1999.
27. Ahlberg J. Candide-3—an updated parameterized face. *Technical Report*, Dept. of Electrical Engineering, Linköping University, Sweden, 2001.
28. Bishop CM. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.
29. Har-peled S, Roth D, Zimak D. Constraint classification for multiclass classification and ranking. In *NIPS-02*. MIT Press, 2003; 785–792.
30. Sim T, Baker S, Bsat M. The CMU pose, illumination, and expression database. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2003; **25**(12): 1615–1618.
31. Kanade T, Cohn JF, Tian Y. Comprehensive database for facial expression analysis, In *Proceedings of Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, Grenoble, France, 2000; 46–53.
32. Huang GB, Ramesh M, Berg T, Learned-Miller E. Labeled faces in the wild: a database for studying face recognition in unconstrained environments. *Technical Report 07-49*, University of Massachusetts, Amherst, October 2007.

AUTHORS' BIOGRAPHIES

Ming Zeng received his B.E. degree in Computer Science from Nanjing University of Posts and Telecommunications in 2008. He is currently working toward the Ph.D. degree in Computer Science at State Key Laboratory of CAD&CG, Zhejiang University. Previously, he was a research intern in Visual Computing group of Microsoft Research Asia. His main research interests include interactive computer vision, geometry acquisition, and 3D reconstruction.

Lin Liang is a researcher in Kent team of Microsoft SBG. She received a B.S. degree in 1998 and a Master's degree in 2001 from the Department of Automation in Tsinghua University. She worked in Microsoft Research Asia in Beijing during 2001–2010. Her main research interests are computer vision, including face alignment, face tracking, face modeling, and machine learning.

Xinguo Liu is a professor in Zhejiang University since 2006. He received a B.S. degree in 1995 and a Ph.D. degree in 2001 from the Department of Applied Mathematics in Zhejiang University. He worked in Microsoft Research Asia in Beijing during 2001–2006. His main research interests are in computer graphics and vision, including geometry processing, realistic and image-based rendering, deformable object, and 3D reconstruction.

Hujun Bao received the B.S. and Ph.D. degrees in Applied Mathematics from Zhejiang University in 1987 and 1993, respectively. Currently, he is a professor and the director of State Key Laboratory of CAD&CG at Zhejiang University. His main research interest is computer graphics and computer vision, including real-time rendering technique, geometry computing, virtual reality, and 3D reconstruction.

 Comp. Anim. Virtual Worlds 2012; **23**:167–178 © 2012 John Wiley & Sons, Ltd.
DOI: 10.1002/cav