DOI:10.1688/JISE.2013.29.6.12

Video-Driven Creation of Virtual Avatars by Component-based Transferring of Facial Expressions^{*}

CHENG-CHIN CHIANG, ZHIH-WEI CHEN AND CHEN-NING YANG Department of Computer Science and Information Engineering National Dong Hwa University Hualien, 974 Taiwan

This paper proposes an efficient and economic video-driven technique that enables the instant creation of a large diversity of virtual avatars and the automatic syntheses of vivid facial animations. The proposed technique addresses the expression transferring problem which transfers a given facial expression of a source human character to the corresponding one of a synthesized avatar. In tackling the expression transferring problem, we propose a component-based approach which is more appealing than the existing approaches which treat the whole face as a single unit for expression transferring. Our approach acquires a much higher diversity in synthesizing virtual avatars and facial expressions by composing the synthesized target face from the facial components of different avatars. The proposed method achieves a good way to transfer the synthesizing parameters acquired from the source human face to those of the target avatar face which complies well with the person-specific characteristics of the target avatar. Additionally, the removal of color inconsistencies among the facial components from different avatars is also well handled. Some experimental results are demonstrated to show that the proposed method can achieve interesting and colorful transfers of facial expressions and synthesize a large diversity of virtual avatars instantly.

Keywords: facial expression synthesis, active appearance model, color correction, facial feature tracking, virtual avatar

1. INTRODUCTION

In recent years, people have experienced the magic attractiveness of creating lifelike and funny virtual characters in many applications, such as film production, computer games, computer-aided instruction, virtual chatting, virtual conferencing, and so on. However, producing these lifelike facial animations for virtual characters requires both the professional skills in art design and some high-end equipments for motion capturing. The expense and time are not affordable for common users. Hence, an economic and efficient way to produce funny virtual characters and vivid facial animations becomes vital and highly desirable.

The video-driven approach [1], also referred to as the performance-driven approach, is an attractive and popular solution for automatic facial animation syntheses. This approach synthesizes a virtual avatar's facial animations which are continuously synchronized with the facial animations of a source human character on a video. Differing from the approach of motion capturing, the video-driven approach does not require the human character to wear any marker on his or her face. Instead of exploiting high-end motionsensing hardware, the video-driven approach uses only a low-end camera and an intelli-

Received May 16, 2011; revised November 11, 2011 & February 22, 2012; accepted March 3, 2012.

Communicated by Tyng-Luh Liu.

^{*} This work was supported by the National Science Council of Taiwan under the Grant No. 97-2221-E-259-028-MY3 and 100-2221-E-259-032-.

gent computer vision algorithm to automatically track the human facial animations. Some modeling parameters are then acquired from the tracked facial animations and transformed to those of the virtual avatar. Consequently, the facial animations can be synthesized and synchronized on the virtual avatar from the transformed parameters accordingly. We call this transferring of facial animation parameters an expression transfer process between the source human character and the virtual avatar.

A common method to do the facial expression transfers is to treat the whole face as a unit. This way would restrict the diversity of the synthesized faces because only a very limited number of avatars can be synthesized. On the contrary, if we could transfer the expressions of individual facial components, such as eyes, nose, and mouth, of the source character to those of many different avatars, then we can compose many novel 'hybrid' virtual avatars by combining the transferred facial components of different avatars into a target face. Obviously, this component-based synthesis of virtual avatars would significantly enlarge the diversity of synthesizable faces. In addition, this component-based approach also relieves the animation designers from the heavy load to prepare a large database of face models for many avatar candidates to be synthesized. A large number of interesting novel virtual avatars can be synthesized under the availability of only a few avatar candidates' face models.

The idea of the component-based syntheses of facial expressions and virtual avatars may be somewhat intuitive. Nonetheless, this approach involves two non-trivial technical issues. One is to find the transformation between the expression parameters of the source human character and those of the target avatar. Directly copying the expression parameters of the source human character to those of the target avatar can cause unnatural or unmatched facial animations if the source human and the target avatar have very different face geometries. A good transformation should output an expression that well comply with the person-specific characteristics of the target avatar. The other issue is to remove the inconsistencies of color tones among the synthesized facial components if they come from different avatar candidates. The removal of color inconsistencies demands a color correction method which can unify the color tones of different facial components.

Follow the merit of the component-based syntheses of facial expressions, a method for facial expression transferring is proposed in this paper. We exploit the *active appearance model* (AAM) [2-5] to automatically track facial features on each video frame. AAM provides good parametric representations for both shapes and textures of faces and facial components. The two technical issues for the component-based synthesizing approach mentioned above are also well tackled in the proposed method. Fig. 1 illustrates the process flow of the proposed component-based transferring of facial expressions. The process comprises five stages. The major tasks performed in each stage are briefly described in the following.

- 1. *Model training stage*: to build the required person-specific AAMs from the face samples of the source human characters and target avatars;
- 2. *Facial component tracking/extraction stage*: to acquire the shape deformation of the source human's facial components by AAM face tracking;
- 3. *Shape transferring stage*: to infer the shape deformation of each facial component on the target face from the acquired shape deformation on the corresponding facial component of the source human and then compose a synthesized shape for the target



Fig. 1. The process flow of the proposed method.

facial component;

- 4. *Appearance transferring stage*: to synthesize the textures for the target facial components;
- 5. *Color correction stage*: to remove the color inconsistencies among the synthesized facial components and then map the corrected target textures onto the synthesized shapes of target facial components.

The rest of this paper is organized as follows. In Section 2, we survey the related work for video-driven syntheses of facial animation. Section 3 gives a short review on the mathematical framework and operating principles of the AAM. The proposed component-based facial expression transferring method are presented in Section 4. To examine the performance of the proposed method, Section 5 demonstrates some synthesizing results and compares the results with those of other related work. Finally, Section 6 ends this paper with some concluding remarks.

2. RELATED WORK

Facial component tracking/extraction and facial expression syntheses are two core tasks involved in video-driven expression transferring. For facial components tracking/ extraction, a commonly adopted approach is the feature-based approach which detects and tracks several salient landmarks on a face. Pantic and Rothkrantz [6] specified more than 30 facial features on frontal- and profile-view faces to characterize facial actions for different facial expressions. Zhang et al. [7] also proposed a probabilistic framework for analyzing and synthesizing facial expressions by locating facial landmarks defined in MPEG-4 facial animation parameters (FAPs). Some specific low-level image feature detectors are usually designed to locate the salient facial landmarks on a face. Mahoo [8] provides a good table to summarize the related methods. For example, Feris and Junior [9] used the Gabor wavelet feature and a radial basis function (RBF) network to locate and track salient facial points. McKenna [10] integrated the Gabor wavelet feature with a shape model, called Point Distribution Model (PDM), for tracking rigid and non-rigid facial motions. Other possible features might include colors, edges, corner points, and contours [11-16]. However, none of the proposed image feature detectors is generally effective for detecting all kinds of facial components due to the different appearances of different facial components. The changing head poses and variational facial expressions would also impose high challenges to the automatic positioning of these specifically-defined features.

Extracting faces or facial components demands a good parametric representation to characterize both the shapes and textures. For example, the parabolic contours of lips, eyes and mouths are usually parameterized by the coefficients of parabolic functions or multiple independent polynomial curves [17-20]. These coefficients can be estimated from the located positions of several landmarks on the contours of the facial components. The facial components can then be extracted after deriving these coefficients. However, the parabolic or polynomial functions are not general enough to characterize all possible deformation that might appear on these facial components. Pantic and Rothkrantz [6] exploited different parametric representations for extracting different facial components. The heterogeneous parameterizations would thus increase the complexity of the system design.

The active appearance model (AAM) [3] is a generic and powerful model to parameterize both the shapes and textures of visual objects. An AAM approximates the shape and texture of a visual object by using a set of basis shapes, called *eigenshapes*, and a set of basis textures, called *eigentextures*, respectively. The AAM has been successfully applied to a wide range of applications in object tracking and object segmentation. Orozco *et al.* [21] exploited an AAM to track eyelids, eyebrows, and lips at a real time speed. Kobayashi and Satake [22] proposed a person-independent AAM for tracking human faces. Jiang [23] used an AAM to track the facial landmarks defined in MPEG-4 FDPs. The AAM needs no predefined low-level image feature detectors for tracking facial component. Hence, different facial components can be effectively tracked by the generic tracking algorithm of the AAM. However, one major weakness of AAMs is the time-consuming efforts for the model training. The training process requires the preparation of a collection of training face images and demands tedious manual positioning of landmarks on each training face image.

As to the synthesizing methods, Vetter and Blanz et al. [24] presented an approach based on the three-dimensional morphable model (3DMM) [25]. The 3DMM exploits a 3D parametric face model that provides control over the deformation of facial geometries and face textures. The major disadvantage of using a 3DMM is that a collection of 3D laser range scan images is required for the 3D modeling. This requirement significantly increases the cost and overhead for model preparation. Macedo et al. [26] proposed a 2D approach which was based on a multilinear AAM. Based on multilinear analysis [27], their 2D approach constructed two 3rd-order tensors, one for the shape and the other for the texture, to store the AAM representations. The AAM representations in each tensor were structured according to three modes corresponding to the person identity (17 subjects), expression type (7 expressions), and AAM coefficients, respectively. Corresponding to each tracked face, two parameter vectors, which characterized respectively the person identity and the expression type of this face, were retrieved by a tensor projection. Therefore, four parameter vectors can be obtained from the tensors given a pair of specified source face and target face. From these four parameter vectors, one can combine the two parameter vectors, which correspond respectively to the target person identity and the source face expression, to output the desired expression for the target person. One major weakness of this approach is that the synthesized facial expression types and person identities are limited to only a few pre-stored in the tensor representations.

On transferring an input facial expression of a source person to the face of a target avatar, directly applying the source person's facial deformation to the target avatar's face may lose the person-specific characteristics of the target avatar if their faces differ much in geometry. Theobald *et al.* [28] addressed this problem by computing the pairwise correlations between the eigenshapes and eigentextures of the source person and the target avatar. The computed correlations were introduced into the transformation of deformation between the source face and the target face such that the person-specific characteristics of the target avatar can be well preserved. However, since the transformation is derived from the person-to-person correlations, the diversity of the transferred facial expressions is thus limited by the number of person candidates and avatar candidates.

Our work presented in this paper makes some improvements over the existing methods. Firstly, we propose to use the component-based approach for facial expression transferring. This approach can significantly increase the variety of the synthesized facial expressions and virtual avatars without requiring the preparation of the parametric face models for many candidate characters or avatars. Secondly, the transferred facial expressions are not limited to only a few common facial expressions or only a few person identities. Instant creation of different facial expressions of novel 'hybrid' avatars becomes possible. Thirdly, the synthesized target facial expressions well comply with the target avatar's person-specific characteristics by component-to-component correlations, instead of person-to-person correlations.

3. FACIAL COMPONENT TRACKING AND EXTRACTION BY ACTIVE APPEARANCE MODELS

3.1 Active Appearance Model

The AAM, proposed by Cootes et al. [3], offers a parametric representation for both

object shapes and object textures. The shape of a face comprises a set of predefined triangles whose vertices are from a set of specified facial landmarks. Suppose that the *i*th landmark on a face image is located at (x_i, y_i) , for $1 \le i \le L$. Then, the vector $\mathbf{s} = [x_1, y_1, ..., x_i, y_i, ..., x_L, y_L]^t$, in which the triangle vertices are arranged in a specific order, serves as a shape representation of the face. To acquire a more compact representation for the long vector *s*, the principal component analysis (PCA) [29] is performed to derive a set of orthogonal basis vectors, say $\{\mathbf{\hat{s}}_i | \mathbf{\hat{s}}_i \in \mathbb{R}^{2L}, 1 \le i \le M\}$. These basis vectors are called the *eigenshapes*. The shape vector of each given face, say \mathbf{s} , can thus be approximated with the linear combination of these *M* eigenshapes by

$$\mathbf{s} = \overline{\mathbf{s}} + p_1 \mathbf{\hat{s}}_1 + \ldots + p_M \mathbf{\hat{s}}_M,\tag{1}$$

for the coefficients $p_1, p_2, ..., p_M$. We call the parameter vector $\mathbf{p} = [p_1, p_2, ..., p_M]^t$ the shape parameter vector of \mathbf{s} .

Similarly, for representing a face texture, a texture vector can be obtained by arranging the 2D pixel values of a face image into a 1D vector. Also by doing the PCA on a set of face textures, the original texture space can be reduced to a compact representation in terms of a set of orthogonal eigentextures, say $\{\hat{a}_1, \hat{a}_2, ..., \hat{a}_M\}$. In other work, each given face texture can be approximated by

$$\mathbf{a} = \bar{\mathbf{a}} + \lambda_1 \hat{\mathbf{a}}_1 + \ldots + \lambda_{M'} \hat{\mathbf{a}}_{M'},\tag{2}$$

for some coefficients $\lambda_1, \lambda_2, ..., \lambda_M$. The vector $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, ..., \lambda_M]^t$ is called the texture parameter vector of the given face texture **a**. In our targeted problem of facial synthesis, we build a personalized AAM for each human person and avatar. Through the statistical PCA training process, these person-specific AAMs can well capture the person-specific characteristics, including shape geometries and texture appearances, of the persons and avatars.

3.2 Face Tracking and Facial Component Extraction Using AAMs

Tracking a face with an AAM is actually a model fitting process. The model fitting process iteratively updates the shape parameter vector and texture parameter vector to minimize the difference between the reconstructed facial appearance and the observed face appearance. Each observed face shape may present two types of deformation. One is the piecewise local warping, $W(\mathbf{x}; \mathbf{p})$, which encodes the local displacement of face pixels. This warping function can be characterized by the shape parameter \mathbf{p} . The other type of deformation is induced by the global rotation and scaling of the face. This deformation is usually characterized by an affine transformation, $N(\mathbf{x}; \mathbf{p})$, where \mathbf{q} denotes the vector of affine parameters. Matthews and Baker [30] formulated the AAM's fitting process as a process to minimize the following error residual

$$\sum_{\mathbf{x}\in\overline{\mathbf{s}}} [E(\mathbf{x})]^2 = \sum_{\mathbf{x}\in\overline{\mathbf{s}}} [\mathbf{a}(\mathbf{x}) - I(N(W(\mathbf{x};\mathbf{p});\mathbf{q}))]^2,$$
(3)

Where $\mathbf{a}(\mathbf{x})$ denotes the reconstructed pixel value at the position \mathbf{x} on the reconstructed

texture and $I(\mathbf{x})$ denotes the pixel value at the position \mathbf{x} on the observed face image I. Substituting the reconstructed texture $\mathbf{a}(\mathbf{x})$ with the right-hand side of Eq. (2) leads to

$$\sum_{\mathbf{x}\in\overline{\mathbf{s}}} [E(\mathbf{x})]^2 = \sum_{\mathbf{x}\in\overline{\mathbf{s}}} [\overline{\mathbf{a}}(\mathbf{x}) + \sum_{i=1}^{M} \lambda_i \hat{\mathbf{a}}_i(\mathbf{x}) - I(N(W(\mathbf{x};\mathbf{p});\mathbf{q})]^2.$$
(4)

The parameters include λ_i , **p**, and **q** can be estimated by the efficient inverse compositional AAM (ICAAM) algorithm proposed by Matthews and Baker. Readers who are interested in the details can refer to the work of Matthews and Baker [30].

The AAM face fitting requires a proper initialization on the position and size of the face. To this end, we employ the Adaboost face detector [31] for the automatic initialization on the first frame of a video. For the subsequent frames, each frame uses the tracked result on the preceding frame as the initialization. Once the face fitting of the AAM fails on a certain frame, the Adaboost face detector starts again to acquire a new initialization. If the AAM fitting still fails with the new initialization, then the current frame is skipped directly. In this way, our AAM fitting process needs no manual assistance.

Since the proposed method is a component-based approach, we partition the AAM of a face into facial components including eyes, nose, mouth, and face silhouette (also called bare face in this paper), as shown in Fig. 2. Once the AAM finishes the model fitting of a given face, the shape and texture of each facial component can be easily segmented from the fitted face.



Fig. 2. The polygonal meshes to represent the shape of a face and facial components. The left image is the shape of the whole face. The middle image contains the shapes of eyes, nose, and mouth. The right image is the shape of the face silhouette.

4. COMPONENT-BASED FACIAL EXPRESSION TRANSFER AND VIRTUAL AVATAR CREATION

This section presents the three steps involved in the proposed method for facial expression transfer and virtual avatar creation.

4.1 Component-based Transferring of Shapes and Textures by AAMs

Let $\mathbf{S} = \{\hat{\mathbf{s}}_1, \hat{\mathbf{s}}_2, ..., \hat{\mathbf{s}}_m\}$ and $\mathbf{A} = \{\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2, ..., \hat{\mathbf{a}}_{m'}\}$ contain the eigenshapes and eigentextures of an AAM, respectively. Meanwhile, the mean shape and the mean texture are $\overline{\mathbf{s}}$ and $\overline{\mathbf{a}}$, respectively. Given a source human's face image, the AAM can acquire the shape parameter vector $\mathbf{p} = [p_1, p_2, ..., p_m]^t$ and the texture parameter vector $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, ..., \lambda_{m'}]^t$ to approximate the shape and texture after the fitting process. We define the

approximating shape function $\mathcal{S}()$ and the approximating texture function $\mathcal{A}()$ as

$$\mathcal{S}(\mathbf{p} = [p_1, p_2, \dots, p_m]^i; \overline{\mathbf{s}}, \mathbf{S}) = \overline{\mathbf{s}} + \sum_{i=1}^m p_i \hat{\mathbf{s}}_i, \text{ and}$$
(5)

$$\mathcal{A}(\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \dots, \lambda_{m'}]^{t}; \overline{\mathbf{a}}, \mathbf{A}) = \overline{\mathbf{a}} + \sum_{i=1}^{m'} \lambda_i \hat{\mathbf{a}}_i.$$
(6)

Hence, with respect to the mean face shape and mean face texture, the local deformation in face shape and texture appeared on the input human face are

$$\Delta S(\mathbf{p}; \bar{\mathbf{s}}, \mathbf{S}) = S(\mathbf{p}; \bar{\mathbf{s}}, \mathbf{S}) - \bar{\mathbf{s}} = \sum_{i=1}^{m} p_i \hat{\mathbf{s}}_i \text{ and}$$
(7)

$$\Delta \mathcal{A}(\boldsymbol{\lambda}; \overline{\mathbf{a}}, \mathbf{A}) = \mathcal{A}(\boldsymbol{\lambda}; \overline{\mathbf{a}}, \mathbf{A}) - \overline{\mathbf{a}} = \sum_{i=1}^{m'} \lambda_i \hat{\mathbf{a}}_i,$$
(8)

respectively. The two types of deformation jointly reflect different visual appearance such as shading, illumination, and expressional variations on the human face. To transfer the similar visual appearance of the human's face to the synthesized target avatar's face, both the shape deformation and the texture deformation must be transformed to the face of the target avatar.

For ease of reference, we denote the mean shape of a specific facial component c of a person J by $\overline{\mathbf{s}}_{J}^{c}$. The set $\mathbf{S}_{J}^{c} = \{\hat{\mathbf{s}}_{J,1}^{c}, \hat{\mathbf{s}}_{J,2}^{c}, \dots, \hat{\mathbf{s}}_{J,m_{J}}^{c}\}$ contains the m_{J} eigenshapes of the person-specific AAM built for the facial component c of the person J. Let I and O be respectively the identities of the source human and the target avatar. Thus, $\Delta S(\mathbf{p}_{I}^{c}; \mathbf{\bar{s}}_{I}^{c}, \mathbf{S}_{I}^{c})$ denotes the local shape deformation on the facial component c of the source person I, where \mathbf{p}_{t}^{c} is the shape parameter vector of the facial component c acquired from the AAM fitting. Similarly, $\Delta S(\mathbf{p}_{O}^{c}; \mathbf{\bar{s}}_{O}^{c}, \mathbf{S}_{O}^{c})$ denotes the transferred local shape deformation on the facial component c of the target avatar O, where \mathbf{p}_{o}^{c} is the shape parameter vector to be computed from the transfer process. Directly copying $\Delta S(\mathbf{p}_{l}^{c}; \mathbf{\bar{s}}_{l}^{c}, \mathbf{S}_{l}^{c})$ to $\Delta S(\mathbf{p}_{o}^{c}; \mathbf{\bar{s}}_{o}^{c}, \mathbf{S}_{o}^{c})$ is inadequate if the corresponding facial components of subject I and subject O have very different geometries. For example, if the target avatar's mouth is much larger than the source person's mouth, then a large scale of shape deformation on the source person's mouth may drive only unnoticeable mouth motion on the target avatar's face. This improper transformation of deformation would result in unnatural or unmatched facial animation on the avatar's face. Therefore, the correlation between the person-specific facial characteristics of the source person and the target avatar must be considered during the deformation transformation.

The proposed method have two dedicated designs for transferring the shape deformation. Firstly, we incorporate a person-specific scaling factor into the local shape deformation. The scaling factor accounts for the size difference of the corresponding facial components of different subjects. Let \mathbf{x}_0 and \mathbf{y}_0 be respectively the shape center of the corresponding facial components of the source human and the target avatar. Let each shape have L landmarks, a landmark \mathbf{x} on the source human's facial component corresponds to the landmark \mathbf{y} on the target avatar's facial component. This correspondence is denoted by $\mathbf{x} \equiv \mathbf{y}$. Then, the scaling factor is computed by

$$\alpha(I,O)^{c} = \frac{1}{L} \sum_{\mathbf{x} \in \mathbf{\bar{s}}_{0}^{c} \land \mathbf{x} \in \mathbf{y}} \frac{\|\mathbf{x} - \mathbf{x}_{0}\|}{\|\mathbf{y} - \mathbf{y}_{0}\|},\tag{9}$$

where $||\mathbf{u} - \mathbf{v}||$ denotes the distance between two points \mathbf{u} and \mathbf{v} . The scaling factor $\alpha(I, O)^c$ in Eq. (9) computes the averaged size ratio between the corresponding facial components of the subject *I* and the subject *O*. With the scaling factor $\alpha(I, O)^c$, the local shape deformation of the source facial component must be scaled by

$$\Delta \mathcal{S}(\mathbf{p}_{I}^{c}; \overline{\mathbf{s}}_{I}^{c}, \mathbf{S}_{I}^{c}) \leftarrow \alpha(I, O)^{c} \Delta \mathcal{S}(\mathbf{p}_{I}^{c}; \overline{\mathbf{s}}_{I}^{c}, \mathbf{S}_{I}^{c}), \tag{10}$$

when transformed to the target avatar's facial component.

Our second design for transforming the local shape deformation is to approximate the scaled local shape deformation on the source facial component by the eigenshapes of the target avatar's facial component. Due to the statistical nature of the PCA, the eigenshapes of each facial component characterize statistically the person-specific shape of the facial component. Therefore, using the eigenshapes of the target avatar to approximate the shape deformation should comply well with the statistical characteristics of the target avatar. The deformation approximation can be written by

$$\sum_{i=1}^{m_O} p_{O,i}^c \hat{\mathbf{s}}_{O,i}^c \approx \alpha(I,O)^c \Delta \mathcal{S}(\mathbf{p}_I^c; \overline{\mathbf{s}}_I^c, \mathbf{S}_I^c) = \alpha(I,O)^c \sum_{j=1}^{m_I} p_{I,j}^c \hat{\mathbf{s}}_{I,j}^c,$$
(11)

where the term at the left side of the sign " \approx " is the approximated local shape deformation on the target avatar's face, while the term at the right side is the scaled local shape deformation on the source human's face. Since all eigenshapes { $\hat{\mathbf{s}}_{O,i}^{c}$ } $\hat{\mathbf{j}}_{i=1}^{m_{O}}$ of the target avatar O are mutually orthogonal, multiplying both sides of Eq. (11) with the transpose of the eigenshape $\hat{\mathbf{s}}_{O,i}^{c}$ leads to

$$p_{O,i}^{c} \approx \alpha(I,O)^{c} \sum_{j=1}^{m_{I}} p_{I,j}^{c} \left\langle \hat{\mathbf{s}}_{I,j}^{c}, \hat{\mathbf{s}}_{O,i}^{c} \right\rangle, \tag{12}$$

where $\langle \mathbf{s}_1, \mathbf{s}_2 \rangle$ denotes the inner product of two vectors \mathbf{s}_1 and \mathbf{s}_2 . Thus, the shape parameter vector $\mathbf{p}_O^c = [p_{O,1}^c, p_{O,2}^c, \dots, p_{O,m_O}^c]^t$ can be derived from Eq. (12). With the shape parameter vector \mathbf{p}_O^c , the final shape of the target facial component can be synthesized as

$$\mathcal{S}(\mathbf{p}_{O}^{c}; \mathbf{\bar{s}}_{O}^{c}, \mathbf{S}_{O}^{c}) = \mathbf{\bar{s}}_{O}^{c} + \sum_{i=1}^{m_{O}} p_{O,i}^{c} \mathbf{\hat{s}}_{O,i}^{c}.$$
(13)

After obtaining the final shapes of all facial components, we then apply the global affine transform estimated from the ICAAM algorithm to the synthesized target face so that the final target face can have the same head pose as the source human's face.

The transfers of shape deformation reflect mainly the variations of facial expres-

sions, while the transfers of texture deformation exhibit the distinctive facial appearances, including shading, illumination, and person identities. Similarly, the statistical characteristics of eigentextures in each person-specific AAM can well capture the exterior appearances of a subject's facial components. Unlike the transformation of shape deformation, no global scaling factor is introduced into the transfer of texture deformation. By a derivation similar to that in Eqs. (11) and (12), the texture of the target facial component can be transferred by

$$\mathcal{A}(\mathbf{I}_{O}^{c}; \mathbf{\bar{a}}_{O}^{c}, \mathbf{A}_{O}^{c}) = \mathbf{\bar{a}}_{O}^{c} + \sum_{i=1}^{m'} \lambda_{O,i}^{c} \mathbf{\hat{a}}_{O,i}^{c},$$
(14)

where

$$\lambda_{O,i}^{c} \approx \sum_{j=1}^{m_{I}} \lambda_{I,j}^{c} \left\langle \hat{\mathbf{a}}_{I,j}^{c}, \hat{\mathbf{a}}_{O,i}^{c} \right\rangle.$$
(15)

4.2 Component-based Color Correction

As shown in the left image of Fig. 3, the synthesized textures of different facial components may have different color tones if these components come from different candidate avatars. The color inconsistencies among different facial components on the target face need to be removed. To this end, we propose a color correction method to unify the color tones of all facial components. The key idea behind the color unification is to align the color distributions of all facial components to have comparable locations, dispersions, and orientations in the RGB color space. For simplicity, we assume the color distribution of each facial component to be a normal distribution. Let $N_c(\mathbf{\mu}_c = [\mu_r^{(c)}, \mu_q^{(c)}, \mu_h^{(c)}]^t$, Σ_c) denote the normal color distribution of a facial component c. To align the color distribution of a facial component *i* with that of another facial component *j*, $N_i(\mu_i, \Sigma_i)$ must be translated, rotated, and scaled to have the comparable position, dispersion, and orientation as $N_i(\mu_i, \Sigma_i)$. The orientation of a normal distribution $N(\mu, \Sigma)$ is defined by its principal axes, which correspond to the eigenvectors of the covariance matrix Σ . The corresponding eigenvalues determine the dispersion of the distribution along the principal axes. For a color distribution $N_i(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, let $e_k^{(i)} = \left[e_{kr}^{(i)}, e_{kg}^{(i)}, e_{kg}^{(i)}\right]^T$ be the k^{th} $(1 \le k \le 3)$ eigenvector and $\lambda_k^{(i)}$ be the corresponding eigenvalue. Similarly, let $e_k^{(j)} = \left[e_{kr}^{(j)}, e_{kg}^{(j)}, e_{kg}^{(j)}\right]^T$ and $\lambda_k^{(j)}$ be respectively the k^{th} ($1 \le k \le 3$) eigenvector and its corresponding eigenvalue of $N_i(\mu_i, \mu_i)$ Σ_{i}). The alignment is accomplished by doing the following geometric transformations:

- 1. a translation, denoted by a transform T_{T_i} , to translate $N(\mu_i, \Sigma_i)$ to the origin of the RGB color space;
- 2. a rotation, denoted by a transform T_{R_i} , to rotate $N(\mu_i, \Sigma_i)$ so that its three principal axes are consistent with the R, G, and B axes of the RGB color space;
- 3. a scaling, denoted by a transform $T_{S_{i\to j}}$, to resize $N(\mu_i, \Sigma_i)$ along the principal axes so that $N(\mu_i, \Sigma_i)$ and $N(\mu_i, \Sigma_i)$ have the same dispersion along the principal axes;
- 4. a rotation, denoted by a transform T_{R_i} , to rotate $N(\mu_i, \Sigma_i)$ so that both $N(\mu_i, \Sigma_i)$ and $N(\mu_i, \Sigma_i)$ have the same orientation;
- 5. a translation, denoted by a transform T_{T_j} , to translate $N(\mu_i, \Sigma_i)$ to the center of $N(\mu_j, \Sigma_j)$.

In summary, the above transformations for unifying $N(\mu_i, \Sigma_i)$ with $N(\mu_j, \Sigma_j)$ can be represented with a composite transform $T_{i \rightarrow j}$ where

$$T_{i \to j} = T_{T_j} \cdot T_{R_j} \cdot T_{S_{i \to j'}} \cdot T_{R_i} \cdot T_{T_i}.$$
(16)

The transforms T_{T_i} , T_{R_i} , $T_{S_i \rightarrow j}$, T_{R_j} , and T_{T_j} are derived as

$$\begin{split} T_{T_{i}} &= \begin{bmatrix} 0 & 0 & 0 & -\mu_{r}^{(i)} \\ 0 & 0 & 0 & -\mu_{b}^{(i)} \\ 0 & 0 & 0 & -\mu_{b}^{(i)} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \ T_{R_{i}} &= \begin{bmatrix} e_{1r}^{(i)} & e_{1g}^{(i)} & e_{1b}^{(i)} & 0 \\ e_{2r}^{(i)} & e_{2g}^{(i)} & e_{2b}^{(i)} & 0 \\ e_{3r}^{(i)} & e_{3g}^{(i)} & e_{3b}^{(i)} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \ T_{S_{i\to j}} = \begin{bmatrix} \sqrt{\lambda_{1}^{(j)} / \lambda_{1}^{(i)}} & 0 & 0 & 0 \\ 0 & \sqrt{\lambda_{2}^{(j)} / \lambda_{2}^{(i)}} & 0 & 0 \\ 0 & 0 & \sqrt{\lambda_{3}^{(j)} / \lambda_{3}^{(i)}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\ T_{R_{j}} &= \begin{bmatrix} e_{1r}^{(j)} & e_{1g}^{(j)} & e_{1b}^{(j)} & 0 \\ e_{2r}^{(j)} & e_{2g}^{(j)} & e_{2b}^{(j)} & 0 \\ e_{3r}^{(j)} & e_{3g}^{(j)} & e_{3b}^{(j)} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \ \text{and} \ T_{T_{j}} &= \begin{bmatrix} 0 & 0 & 0 & \mu_{r}^{(j)} \\ 0 & 0 & 0 & \mu_{b}^{(j)} \\ 0 & 0 & 0 & \mu_{b}^{(j)} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \end{split}$$

Once the transformation $T_{i\to j}$ is computed, the color correction of a given color tuple (r_i, g_i, b_i) in $N(\mu_i, \sum_i)$ can be done by $[r_j, g_j, b_j, 1]^t = T_{i\to j} \cdot [r_i, g_i, b_i, 1]^t$, where (r_i, g_i, b_i) is the corrected color tuple which is consistent with the target color distribution $N(\mu_i, \sum_i)$. The right image in Fig. 3 demonstrates the result of applying the color correction on the left image in Fig. 3. Both the color distributions of the nose and mouth are aligned with that of the face silhouette. Alternatively, we can specify the skin color distribution of any specific person as the target color distribution to be aligned with.

During the component-based synthesis of target faces, some seaming edges may appear between neighboring facial components. These seaming edges can be removed by blending the pixel colors around the component boundaries. Two blending masks, $W_b(\mathbf{x})$ and $W_c(\mathbf{x})$ shown in Fig. 4, are exploited. The intensity value of each pixel on both masks stores the weight for pixel blending. The blending mask W_b , which is used for the bare face, overlaps slightly with the blending mask $W_c(\mathbf{x})$, which is used for other facial components. The weight values on both masks can be pre-computed in advance. The blended intensity value of a pixel x is computed by:



Fig. 3. An example of color correction. The left image is the face before color correction, while the right image is the face after color correction.



Fig. 4. The blending masks for removing seaming edges along the boundaries of facial components. The left image is the blending mask $W_c(\mathbf{x})$ for the nose, mouth, and left/right eyes. The right image is the blending mask $W_b(\mathbf{x})$ for the bare face.

$$\mathcal{A}(\mathbf{x}) = W_b(\mathbf{x})\mathcal{A}(\mathbf{x}, \mathbf{p}^b, \mathbf{\bar{s}}^b, \mathbf{A}^b) + W_c(\mathbf{x})\mathcal{A}(\mathbf{x}, \mathbf{p}^c, \mathbf{\bar{s}}^c, \mathbf{A}^c),$$
(17)

where $W_b(\mathbf{x}) + W_c(\mathbf{x}) = 1$ for each pixel \mathbf{x} . In implementation, the mask $W_c(\mathbf{x})$ can be simply obtained by doing a 7×7 Gaussian filter on a binary mask image enclosing all facial components and then normalizing the pixel values into the interval [0, 1]. With $W_c(\mathbf{x})$, the mask $W_b(\mathbf{x})$ can be derived accordingly.

Another problem in our face syntheses is the handling of missing-texture areas which usually appear in the mouth cavity and eyeballs. Since no AAMs are built for the tongue, teeth, and eyeballs of each subject, the textures of these components cannot be synthesized. Our simple solution is to use the textures of the tongue, teeth and eyeballs segmented from the source human's face directly. These textures can be easily extracted once the AAM face fitting is accomplished. After synthesizing the textures of all facial components, we map the synthesized textures onto the transferred shapes of the target facial components to compose the final target face.

5. EXPERIMENTAL RESULTS

In our experiments, there are four candidate subjects, including two real humans and two artificial characters, in the subject database. To train the person-specific AAM for each subject, we collect 13 to 17 face images with different head poses and facial expressions. The image samples of the two real humans are captured from a low-cost web camera at the resolution of 640×480 pixels. The face sizes are around 220×320 pixels. The samples of the two artificial characters, Shrek and Gollum (a character in the movie "The Ring"), come from the video clips of the movies.

5.1 Results for Person-to-Person Expression Transfers

Like other existing methods, the proposed method can accomplish the person-toperson expression transfers. A person-to-person expression means that all of the target avatar's facial components come from a single subject candidate. Hence, both the source human character and the target avatar are subject candidates in the database. On doing each person-to-person expression transfer, we can have two different resultant faces. One is obtained by a *face reshaping step* which simply warps the source human's face texture onto the synthesized face shape of the target avatar. The resultant face still looks like the source human, except that the face shape has been changed. The other resultant face is obtained by an *expression cloning step* which warps the synthesized avatar's face texture onto the synthesized avatar's face shape. The expression cloning step outputs the final result of the person-to-person expression transfer.

In Fig. 5, several results of face reshaping are shown. The source faces on video input, which are listed in the first row of Fig. 5, have different facial expression. The resultant target faces in the second, third, and fourth rows show the reshaped faces with respect to different synthesized avatar's face shapes. The demonstrated results show that some interesting novel avatars can be created from the face reshaping step.

For expression cloning, the resultant faces in Row 2 through Row 4 of Fig. 6 show the results of cloning the four source human's expressions shown in the top row to three different target avatars, including the subjects Joker, Gollum, and Shrek, respectively. Unlike some other existing methods, our cloned facial expressions are not limited to only a few specific expressions. Any expression can be cloned as long as the face can be well fitted by the AAM. We have tested many different exaggerative expressions and the AAM tracking results are pretty good for frontal faces. The synthesizing speed is around 5~7 frames/sec on a Intel Core i7-940 CPU. The visual qualities of the synthesized results are good enough for some facial animation applications of virtual chatting and virtual conferencing.



Fig. 5. Synthesized results after person-to-person face reshaping. Row 1 shows the source facial expressions and Row 2 through Row 4 show the reshaped faces using the face shapes of other people.



Fig. 6. Synthesized results after person-to-person expression cloning. Row 1 shows the source facial expressions. Row 2 through Row 4 show the cloned expressions on the faces of "Joker", "Gollum", and "Shrek", respectively.

5.2 Syntheses for Component-based Expression Transfers

The major drawback of the person-to-person expression transfers is that the number of subjects in the database can limit the diversity of synthesizable avatars. The componentbased expression transfers can significantly enlarge the limited diversity even though only a small number of subjects are available in the database.

For each component-based expression transfer, we also examine both the results of component-based face reshaping and the component-based expression cloning. The component-based face reshaping warps the facial component textures of the source input face onto the synthesized shapes of the corresponding facial components of target avatars. Note that different subjects can be specified as the target avatars for reshaping different facial components. Hence, the final synthesized avatar can be a novel subject which does not exist in the database. For example, we may synthesize a novel target avatar by using the shapes of Joker's eyes, Shrek's nose, and Gollum's mouth as the respective target shapes of the composed facial components. Fig. 7 illustrates the reshaped results. The

faces in Column 2 through Column 5 are the reshaped results of the respective source face textures given in Column 1. As demonstrated by the results, the component-based face reshaping synthesizes many funny expressions on novel avatars.





Fig. 7. Some novel faces composed by component-based face reshaping. The reshaped facial components include the eyes, the noses, and the mouths, but not the face silhouette. Column 1 shows the source faces with different expressions. Column 2 – Column 5 show the corresponding reshaped faces. Note that the reshaped composing facial components on each target face are from different subjects.



For the component-based expression cloning, Fig. 8 demonstrates some results of the component-based expression cloning. Again, the results show that colorful facial expressions and avatars can be synthesized even though we have only four subject candidates for syntheses. Compared with the component-based face reshaping, the component-based expression cloning further increases the diversity of the synthesizable avatars to a large extent. The demonstrated results have shown that the proposed componentbased approach can facilitate the video-driven instant creation of virtual avatars.

To examine the effectiveness of the proposed color correction method for the component-based facial syntheses, we show some results in Fig. 9. The middle row and the bottom row of this figure show the transferred facial expressions by synthesizing different target avatars' facial components before and after the color correction, respectively. By aligning the color distributions of the nose and the lip with that of the bare face on each face, the color inconsistencies among the facial components disappear from the target faces.

In the proposed method, the AAM plays a key role in the following aspects. A successful expression transfer strongly relies on the success of the AAM face fitting. In

our tests, the AAM fits the faces very well for frontal or near-frontal faces. For the applications of virtual chatting or virtual conferencing, assuming stable head poses is generally reasonable. The rate of tracking failure for 600 testing images from five video clips is below 5%. Any facial expression is allowable as long as no facial component is vague or absent on the source input face.



Fig. 9. Results of color correction on facial components. The top row shows the source faces. The middle row shows the synthesized results before color correction. The bottom row shows the synthesized results after color correction.



Fig. 10. Comparison between the synthesized faces of the method of Theobald et al. and the proposed method.

5.3 Comparisons with Other Method

Both the method proposed by Theobald *et al.* [28] and the proposed method employ the AAM as the representation model for facial expression syntheses. Particularly, both methods give special consideration to transform the shape and texture deformation of the source face to those of the target face. However, these two methods have two major differences. Firstly, Theobald et al. performs only person-to-person expression transfers, while our method performs not only the person-to-person expression transfers, but also the component-based syntheses of novel avatars. Secondly, in dealing with the deformation transformation for expression transfers, our method further considers the person-specific scaling factor of facial components. Fig. 10 shows some examples to demonstrate the superior synthesizing results after introducing the person-specific scaling factor into the transformation. On the target faces synthesized by our method, the size ratios between different facial components are closer to the ground truths of target avatars. For the results shown in the top two rows of Fig. 10, when transferring the mouth motion of the subject Ning to the subject Gollum, Theobald's method tends to synthesize a smaller mouth on Gollum's face. In contrast, our method synthesizes larger mouths which are more consistent with the true mouth size of Gollum. As shown in the third row of Fig. 10, a similar result can also be seen in transferring the mouth motion of the subject Joker to the subject Shrek. The results show that the proposed method is

superior in transferring the facial expressions between two subjects with unmatched sizes of faces or facial components.

Concerning the task of creating virtual avatars, one might think that the image morphing technique could be also a possible way to produce many different virtual avatars. However, the image morphing technique requires a pair of input facial expression and target facial expression so that the in-between facial expressions can be generated by interpolation. Unfortunately, in the facial expression transferring problem, the target facial expression is unavailable and needs to be synthesized. Therefore, the virtual avatar creation accomplished by the proposed method is totally different from that of image morphing.

6. CONCLUDING REMARKS

This paper presents an economic and efficient video-driven approach for instant creation of virtual avatars and syntheses of facial expressions. Differing from other previous work, the proposed method addresses the problem of facial expression transfers by proposing a component-based approach. With this component-based approach, our method can accomplish both person-to-person expression transfers and component-based creation of novel avatars. The proposed component-based synthesizing method significantly enlarges the diversity of synthesized facial expressions and virtual avatars even though only a few face models of subject candidates are available for syntheses. Technically, the deformation transformation between the source human characters and the target avatars devised in the proposed method well preserves the person-specific characteristics of target avatars. The color correction method also effectively removes the color inconsistencies among the facial components of the target faces.

By exploiting the AAM face fitting, the proposed approach is fully automatic in tracking, extracting, and synthesizing facial expressions. The synthesized facial expressions are also not limited to only a few common expressions. The automated process thus removes the requirement of professional background in art design and the time-consuming labor work in authoring the facial animations. Therefore, the proposed method is highly potential for many avatar-based applications in personal communication, entertainment, and education.

REFERENCES

- 1. F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D. H. Salesin, "Synthesizing realistic facial expressions from photographs," in *Proceeding of the 25th Annual Conference on Computer Graphics and Interative Techniques*, 1998, pp. 75-84.
- 2. T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," in *Proceedings of the 5th European Conference Computer Vision*, 1998, pp. 484-498.
- T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, 2001, pp. 681-685.
- 4. J. Ahlberg,. "An active model for facial feature tracking," *EURASIP Journal on Applied Signal Processing*, Vol. 1, 2002, pp. 566-571.

- C. C. Chiang and Z. W. Chen, "Recognizing partially-occluded faces by recovering normalized facial appearance," *International Journal of Innovative Computing*, *Information and Control*, Vol. 7, 2011, pp. 6219-6234.
- M. Pantic and L. J. M. Rothkrantz, "Facial action recognition for facial expression analysis from static face images," *IEEE Transactions on Systems, Man, and Cybernetics*, Part B, Vol. 34, 2004, pp. 1449-1461.
- Y. Zhang, Q. Ji, Z. Zhu, and B. Yi, "Dynamic facial expression analysis and synthesis with mpeg-4 facial animation parameters," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 18, 2008, pp. 1383-1396.
- M. H. Mahoor, M. Abdel-Mottaleb, and A. N. Ansari, "Improved active shape model for facial feature extraction in color images," *Journal of Multimedia*, Vol. 1, 2006, pp. 21.
- R. S. Feris and R. M. C. Junior, "Tracking facial features using Gabor wavelet networks," in *Proceedings of Brazilian Conference on Computer Graphics, Image Pro*cessing, 2000, pp. 22-27.
- S. J. McKenna, S. Gong, R. P. Wurtz, J. Tanner, and D. Banin, "Tracking facial feature points with gabor wavelets and shape models," in *Proceedings of the 1st International Conference Audio- and Video-Based Biometric Person Authentication*, 1997, pp. 35-42.
- C. C. Chiang, W. K. Tai, M. T. Yang, Y. T. Huang, and C. J. Huang, "A novel method for detecting lips, eyes and faces in real time," *Real-Time Imaging*, Vol. 9, 2003, pp. 277-287.
- E. Saber and A. M. Tekalp, "Frontal-view face detection and facial feature extraction using color, shape and symmetry based cost functions," *Pattern Recognition Letters*, Vol. 19, 1998, pp. 669-680.
- R. S. Feris, T. E. de Campos, and R. M. C. Junior, "Detection and tracking of facial features in video sequences," Lecture Notes in Computer Science, Vol. 1793, 2000, pp. 197-206.
- A. Kapoor and R. W. Picard, "Real-time, fully automatic upper facial feature tracking," in *Proceedings of the 5th International Conference Automatic Face and Gesture Recognition*, 2002, pp. 8-13.
- C. C. Chiang and C. J. Huang, "A robust method for detecting arbitrarily tilted human faces in color images," *Pattern Recognition Letters*, Vol. 26, 2005, pp. 2518-2536.
- C. C. Chiang, M. C. Ho, H. S. Liao, A. Pratama, and W. C. Syu, "Detecting and recognizing traffic lights by genetic approximate ellipse detection and spatial texture layouts," *International Journal of Innovative Computing, Information and Control*, Vol. 7, 2011, pp. 6919-6934.
- A. L. Yuille, P. W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable templates," *International Journal of Computer Vision*, Vol. 8, 1992, pp. 99-111.
- 18. K. Kroschel and M. Heckmann, "Lip parameter extraction for speechreading," in *Proceedings of Conference El. Sprachsignalverarbeitung*, 2002, pp. 58-65.
- 19. B. J. Borgstrom and A. Alwan, "A low complexity parabolic lip contour model with speaker normalization for high-level feature extraction in noise robust audio-visual speech recognition," *IEEE Transactions on Systems, Man and Cybernetics*, Part A,

Vol. 38, 2008, pp. 1273-1280.

- N. Eveno, A. Caplier, and P. Y. Coulon, "Accurate and quasi-automatic lip tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 14, 2004, pp. 706-715.
- J. Orozco, P. Baiget, J. Gonzàlez, and X. Roca, "Eyelids and face tracking in real-time," in *Proceedings of the 6th International Conference on Visualization, Imaging, and Image Processing*, 2006, pp. 1-6.
- 22. A. Kobayashi, J. Satake, T. Hirayama, H. Kawashima, and T. Matsuyama, "Personindependent face tracking based on dynamic AAM selection," in *Proceedings of the 8th IEEE International Conference on Automatic Face and Gesture Recognition*, 2008, pp. 1-8.
- 23. B. Jiang, J. Bu, and C. Chen, "Improving visual awareness by real-time 2D facial animation for ubiquitous collaboration," in *Proceedings of the 11th International Conference on Computer Supported Cooperative Work in Design*, 2007, pp. 151-156.
- 24. T. Vetter and V. Blanz, "A morphable model for the synthesis of 3D faces," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, 1999, pp. 187-194.
- V. Blanz, A. Mehl, T. Vetter, and H. P. Seidel, "A statistical method for robust 3D surface reconstruction from sparse data," in *Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission*, 2004, pp. 293-300.
- 26. E. V. Velho L. Macedo, and I. Brazil, "Expression transfer between photographs through multilinear AAM's," in *Proceedings of the 19th Brazilian Symposium on Computer Graphics and Image Processing*, 2006, pp. 239-246.
- 27. M. A. O. Vasilescu and D. Terzopoulos, "Multilinear analysis of image ensembles: Tensorfaces," Lecture Notes in Computer Science, Vol. 2350, 2002, pp. 447-460.
- B. Theobald, I. Matthews, J. Cohn, and S. Boker, "Real-time expression cloning using active appearance models," in *Proceedings of the ACM International Conference on Multimodal Interfaces*, 2007, pp. 134-139.
- 29. T. W. Anderson, T. W. Anderson, T. W. Anderson, and T. W. Anderson, *An Introduction to Multivariate Statistical Analysis*, Vol. 2, Wiley, NY, 1958.
- 30. I. Matthews and S. Baker, "Active appearance models revisited," *International Journal of Computer Vision*, Vol. 60, 2004, pp. 135-164.
- P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, Vol. 57, 2004, pp. 137-154.



Cheng-Chin Chiang (江政欽) received his Ph.D. degrees in Computer Science and Information Engineering from National Chiao Tung University, in 1993. His research interests include neural networks, pattern recognition and human-machine interface. Dr. Chiang received the honor of Best Long-Term Doctoral Thesis Award from Acer Corporation in 1993 and the honor of Outstanding Young Engineer Award from Chinese Institute of Engineers in 2000. He is currently an Associate Professor of the Department of Computer Science and Information Engineering at National Dong Hwa University, Hualien, Taiwan.



Zhi-Wei Chen (陳志偉) received his B.S. degree in Computer Science and Information Engineering from National Chi Nan University, Taiwan, in 2001. His research interests include computer vision, image processing, and human machine interface. He is currently working towards his Ph.D. degree in the Department of Computer Science and Information Engineering of National Dong Hwa University in Taiwan.



Chen-Ning Yang (楊鎮寧) received the B.S. and M.S. in Computer Science and Information Engineering from National Dong Hwa University in 2006 and 2008, respectively. He is now with the Advantech Corporation, serving as a Software Engineer. His research interests include face tracking, pattern recognition, facial analysis and synthesis.