Dynamic Facial Expression Analysis and Synthesis with MPEG-4 Facial Animation Parameters

Yongmian Zhang, Qiang Ji, Zhiwei Zhu and Beifang Yi

Abstract

This paper describes a faithful reproduction of dynamic facial expressions on a synthetic face model with MPEG-4 facial animation parameters (FAPs) while achieving a very low bit-rate in data transmission. Toward this end, we introduce a coupled Bayesian network (BN) to unify the facial expression analysis and synthesis into one coherent structure. At the analysis end, we cast FAPs and facial action coding system (FACS) into a dynamic BN (DBN) to account for uncertainties in FAP extraction and to model the dynamic evolution of facial expressions. At the synthesizer, a static BN is used to reconstruct FAPs and their intensity. The two BNs are connected statistically through a data stream link. Using a coupled Bayesian network to synthesize the dynamic facial expressions is the major novelty of this work. Our approach has three benefits in synthesizing facial expressions. First, very low bit-rate (9 bytes per frame) in data transmission can be achieved. Second, a facial expression is inferred over time through the DBN so that the perceptual quality of the resulting animation is less affected by the missed or unmeasured FAPs. Third, a more realistic looking facial expression can be reproduced by modeling the dynamic evolution of human expressions. This paper also presents experiments to demonstrate the performance of our system.

Index Terms

Facial expression synthesis, facial animation, Bayesian networks, MPEG-4 standard.

I. Introduction

FACIAL expression synthesis is of interest for many multimedia applications such as human-computer interaction (HCI), entertainment, virtual agents and avatars. Current technologies are still unable to synthesize human expressions in a realistic manner and with crucial emotional contents. Since the MPEG-4 standard [1] will have a crucial role in forthcoming multimedia applications, the facial expression synthesis has gained much interest within the MPEG-4 framework. This also opens a new opportunity for computational study of facial expressions. MPEG-4 provides an alternative way of modeling facial expression and the underlying emotion which are strongly influenced by psychological studies such as

This material is supported in part by a grant from the Air Force Office of Scientific Research under Grant No. F49620-03-0160.

Y. Zhang, Q. Ji and Z. Zhu are with the Department of Electrical, Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180, USA. B. Yi is with the department of Computer Science and Engineering, University of Nevada, Reno, NV 89507, USA

Ekman's facial action coding system (FACS) [2]. FACS has now become the de facto standard in measuring facial expressions.

MPEG-4 standard includes a set of facial definition parameters (FDPs) and a set of facial animation parameters (FAPs). FAPs are used to characterize the movements of facial features defined over jaw, lips, eyes, mouth, nose, cheek. In psychological studies, it is generally believed that the six basic expressions (happiness, sadness, anger, disgust, fear and surprise) can be decomposed into culture and ethnical independent facial action units (AUs) [3]. FAPs are adequate to define the measurement of muscular actions relevant to AUs. Therefore, the six basic facial expressions can be characterized in terms of FAPs. Moreover, FAPs can be placed on any synthetic facial model in a consistent manner with less influence by the inter-personal variations. FDPs are normally transmitted once per session and then followed by a stream of compressed FAPs [1]. The animation of a virtual face is achieved by first transmitting the coded FAPs and then re-synthesizing on the client-side, as shown in Fig. 1. To accommodate very low bandwidth constraint, the FAPs must be compressed so that they can be transmitted in very low bit-rate.

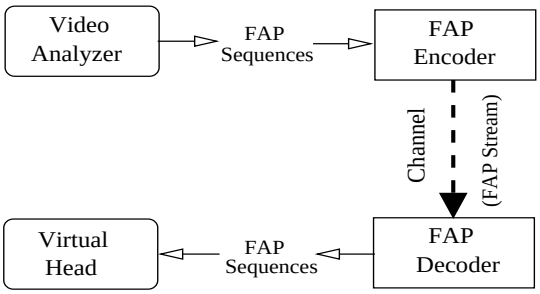


Fig. 1. A general block diagram of a FAP driven facial animation system.

Despite significant progress, current methodologies for facial expression synthesis (see Section II), continue to face several issues that still need to be resolved.

1) Although a high FAP compression efficiency can be achieved for interframe coding with a discrete cosine transform (DCT) technique, DCT involves a large coding delay (temporal latency) that makes it unsuitable for interactive applications. The principal component analysis (PCA) technique can achieve efficient FAP compression for intraframe coding, however, the reconstruction accuracy is often compromised.
2) An automatic video analyzer may often fail to detect feature points for various reasons such as image noise and light change. Consequently, a FAP value between two consecutive frames may not be consistent though the emotional intensity remains unchanged. This may create animation artifacts which affect the perceptual quality of the resulting animation.
3) The intensity of facial expressions reveals the emotional evolution. It is often difficult for machine to extract the subtle variation of facial features. Consequently, a dynamic behavior of human expressions

are difficult to be animated. However, as indicated in [4], temporal course information is necessary for those wanting life-like facial animation.

4) AUs are the linguistic descriptions of facial muscle activities from psychological view. They are often used for grouping the muscle activities in facial animation. FAPs provide a systematic way in defining the measurement of muscular actions. However, there is a lacking of computational model to integrate AUs and FAPs systematically.

This work takes another avenue of approach to address these issues. The proposed approach allows faithful visual reproduction of dynamic human expressions on a synthetic face model using MPEG-4 FAPs, particularly, for low bit-rate interactive applications such as videophone systems and a mobile terminal using a cellular network. Our work has three main contributions. The first contribution is a computational model that systematically integrates FAPs and AUs into a probabilistic framework. We believe that it will be more beneficial to combine AUs and FAPs for facial expression synthesis than using either of them alone. The second contribution is a coupled Bayesian network that allows to unify the facial expression analysis and synthesis into one coherent structure to perform a consistent reasoning and feature fusion. At the analysis end, we cast FAPs and AUs into a dynamic Bayesian network (DBN) to account for uncertainties in FAP extraction and to model the dynamic nature of facial expressions. At the synthesizer, a static Bayesian network (BN) is used to reconstruct FAPs and their intensity. Therefore, the temporal course of a facial expression can be animated. In addition, through the statistical dependencies among FAPs embedded in the BN, robust and accurate reconstruction of the facial expression is possible even in the absence of some FAPs measurements. The third contribution is a very low bit-rate in data transmission to a remote synthesizer for visual reproduction of facial expressions. With a coupled Bayesian network, data communication between analysis end and synthesizer can be implemented as the dependency between two BNs. Therefore, facial expressions are reproduced at a remote synthesizer without recourse to transmitting FAPs. Instead of transmitting a stream of compressed FAPs in MPEG-4 standard, we transmit only 9 bytes of data per frame, i.e., 6 bytes for the probability distribution of the six facial expressions and 3 bytes for face pose (all values can be represented in 8 bits), to a remote synthesizer.

Using a coupled Bayesian network to synthesize the dynamic behavior of facial expressions is the major novelty of this work. The remainder of this paper is organized as follows. The next section reviews related works. Section III provides a brief overview of our system. We present video analysis in Section IV. Our approach in facial expression analysis and synthesis will be covered in Section V and Section VI, respectively. Experiment results are given in Section VII. The final section is conclusion.

II. BACKGROUND

Three areas of research are closely related to the work described in the paper: facial expression analysis, facial expression synthesis and FAP compression. We briefly review the relevant works in each area.

A. Facial Expression Analysis

Automatic facial expression recognition had an early start with static face images [5], [6], [7], [8], [9], [10]. There have been several attempts to recognize facial expressions over time from video sequences. Yacoob and Davis [11] proposed a region tracking algorithm to integrate spatial and temporal information at each frame in an image sequence. Black and Yacoob [12] used local parameterized flow models to identify facial expressions. An affine model and a planar model represent head motion and rotation, and a curvature model represents non-rigid feature motion around the eyebrows and mouth. Essa and Pentland [13] presented a system featuring both facial motion extraction and classification. The facial motion is estimated by using optical flow while facial expression classification is based on the invariance between the motion energy template learned from ideal 2D motion views and the motion energy of the observed image. Oliver et al. [14] applied Hidden Markov model (HMM) and mouth deformation shape to recognize mouth-related expressions. A HMM model is constructed and trained for each expression. The facial expression is identified by computing the maximum likelihood of the input sequence with respect to all trained HMMs. Tian and Kanade [15] used a neural network (NN) approach. Two separate NNs are constructed to recognize the upper face AUs and the lower face AUs. The inputs to the NNs are the parametric descriptions of facial features from multi-state face and facial component models. Zhang and Ji [16] proposed to use a dynamic Bayesian network for modeling the relationships between facial expressions and the facial feature displacements for recognizing the six basic facial expressions.

MPEG-4 visual standard has motivated intensive research in facial feature extraction for facial animation [17], [18], [19], [20], [21]. These works fall in the category of model-based techniques such as deformable template-based model and active appearance model. Substantial efforts in facial expression analysis with MPEG-4 FAPs have been made recently [22], [23], [21]. Among these works, either rule-based technique [22], [23] or HMMs [22], [23], [21] are used. The rule-based approach lacks the expressive power to capture the temporal behaviors and dependencies among facial actions. HMM can model time series with uncertainty, but it cannot represent variables at different levels of abstraction. When HMM is applied to facial expression recognition, each AU must be assigned a specific HMM. Thus, it is difficult to handle the dependencies among facial actions since HMMs are not in one unified model. Additionally, each HMM is associated with only one AU such that it is hard to capture the potential AU combinations since a facial expression often constitutes a combination of AUs.

B. Facial Expression Synthesis

In the area of multimedia, researchers have shown great interest in lifelike animated agents with realistic behavior. Eisert and Girod [24] applied facial expression synthesis to virtual conference, whereby the optical flow is used to estimate the motion information to estimate 17 FAPs for controlling the virtual head. A similar approach is presented by Valente and Dougelay [25]. Chandrasiri et al. [26] presented an Internet chat engine with a client-server paradigm and they intended to reproduce the user's expression at the client side. However, they use a head-phone attached camera to eliminate the face pose, and the expression intensity is interpolated from its pre-stored neutral and apex facial expression. In MPEG-4 facial animation, Tao and Huang [27], Lavagetto and Pockaj [28], Goto et al. [29], Raouzaiou et al. [23], and Kshirsagar et al. [30] proposed a mesh-independent free-form deformation model. The animation of synthetic face is controlled by FAPs. Each FAP defines the animation by specifying feature points and geometric transformation. The facial AUs of the FACS are often used to group the muscle activities in facial animation. For examples, Zhang et al. [31], Terzopoulos and Waters [32] and Waters [33] group muscles into AUs by their position in their facial animation system.

C. FAP Compression

To overcome channel bandwidth limitation, FAPs must be compressed so that they can be transmitted in very low bit rate. The FAP compression can be categorized as intraframe coding and interframe coding.

1) Intraframe Coding: One way to achieve data reduction is to send only a subset of active FAPs to a synthesizer. For a particular facial expression, we only need 27 FAPs associated with 6 basic facial expressions. Additionally, MPEG-4 standard [34] proposed a FAP interpolation table (FIT) that only use a subset of FAPs to interpret the values of other FAPs based on a set of fixed interpolating rules. However, it is generally difficult to adapt such rules to all faces. Tao et al. [35] and Ahlberg and Li [36] use the principal component analysis (PCA) technique. By performing a linear transformation, each FAP is transformed into a new subspace. Although this technique can achieve efficient FAP compression, the reconstruction accuracy is often compromised. It is therefore not suitable for applications with high fidelity in animation is required.

2) Interframe Coding: There are two interframe coding schemes are adopted in MPEG-4 and they are predictive coding (PC) and discrete cosine transform (DCT). In the PC scheme, the difference of FAPs between consecutive frames are encoded and transmitted. Because the differences of FAPs between neighboring frames are usually smaller quantities, fewer bits are needed to represent these differences. The decoded value of each FAP in the previous frame is used to predict its corresponding FAP in the current frame. The predicted error (i.e., the difference between the current FAP value and its prediction)

is coded using an adaptive arithmetic coder. If the FAP sampling rate is relatively high (> 10Hz), DCT technique may be used. By performing DCT in each temporal segment (several consecutive frames), a high compression efficiency can be accomplished. However, due to the significant coding delay, it is unsuitable for interactive applications.

III. System Overview

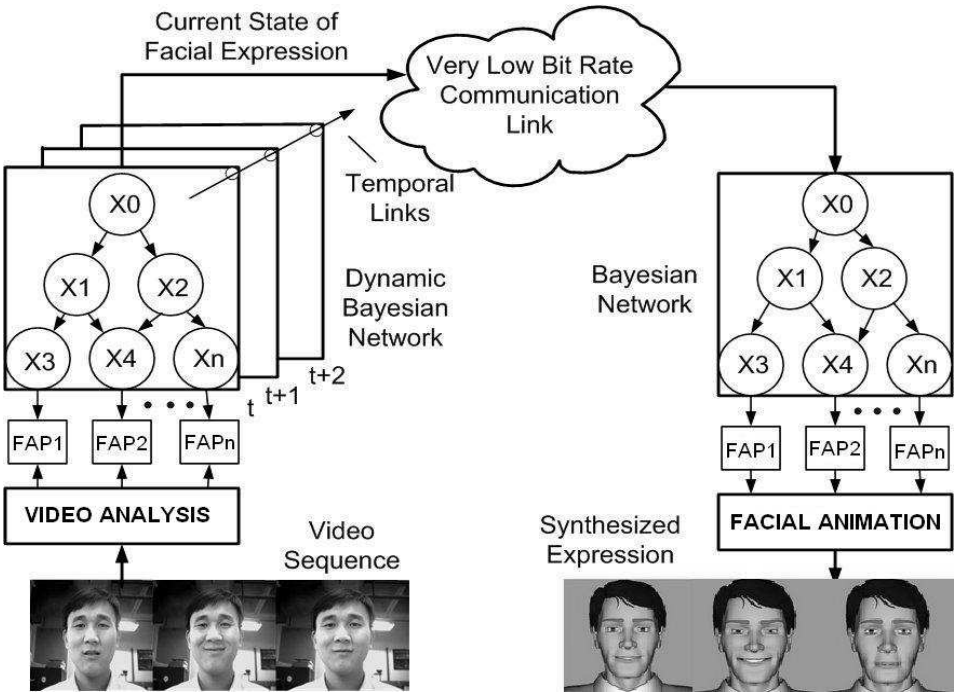


Fig. 2. An overview of our approach for facial expression synthesis, where X_i is a hidden random variable of Bayesian networks, and FAP_i denotes a facial animation parameter (FAP). The dependency between the two top nodes of BNs can be viewed as a data communication link.

Our methods have been integrated into an unified system for FAP extraction, facial expression analysis and synthesis. Fig. 2 depicts the major components of this system. We will introduce each of these components briefly below. In subsequent sections, we describe each of them in more detail.

Video Analysis: Video analysis is to generate FAP values and face pose. The use of facial 3D shape model and eye detection technique makes our facial feature detection and pose estimation robust under the head motion and non-rigid facial expression. The extracted FAPs are the visual evidences for facial expression analysis module.

Expression Analysis: Facial expression analysis is to generate the probability of six facial expressions. This module integrates AUs and FAPs into a dynamic Bayesian network (DBN) to correlate and associate the continual arriving FAPs. The current observed FAPs and previous evidences are fused with Bayesian statistics to generate the probability distribution of six facial expressions.

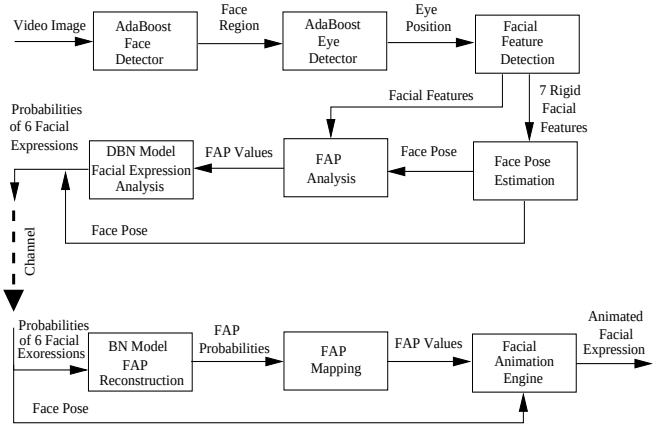


Fig. 3. The detail of system modules and data flow.

FAP Reconstruction: FAPs and their intensity are reconstructed through a static Bayesian network (BN) to provide quantitative information about the motions of evolving facial features. The BN model is coupled with the DBN at analysis end by linking their top nodes so that the probability distribution of six facial expressions at analysis end passes to the synthesizer through a data communication channel.

Facial Animation: This module would use the motion of the reconstructed FAPs to animate its facial model. The interpretation of the reconstructed FAPs and their intensity leads to dynamic muscle parameters. Therefore, a dynamic facial expression can be reproduced on its facial model.

Our system has three major advantages: 1) very low bit-rate in data transmission (only a stream of 9 bytes of data, i.e., 6 bytes for the probability distribution of the six facial expressions and 3 bytes for face pose) can be achieved; 2) a facial expression is recognized over time with the DBN so that the perceptual quality of the resulting animation is less affected by the missed or unmeasured FAPs; 3) a realistic and visually faithful facial expression can be reproduced by modeling the dynamic behavior of human expressions.

Fig. 3 shows the details of system modules and the data flow. The software is developed for FAP extraction and face pose tracking. This software is real time, fully automatic, and applicable to different people. The animation engine of 3D synthetic head model consisting of 3000 vertexes and 3200 polygons is developed. We utilize Intel's Probabilistic Networks Library[1] to build BN facial expression models. The DBN facial expression model is interfaced with our facial feature extraction software to perform automatic facial expression analysis. The BN model is interfaced with our 3D facial model to perform FAP reconstruction and facial expression animation. Data from facial expression analysis is passed on to the synthesizer through TCP/IP protocol, exactly as would be performed in an actual application.

[1]http://www.intel.com/research/

IV. VIDEO ANALYSIS

Developing an automatic video analyzer is still a non-trivial task. FAP extraction in many facial animation systems needs manual involvement. In this section, we first give a brief introduction to the MPEG-4 visual standard with just enough detail that will enable us to apply it to our work. Then we describe our approach in FAP extraction and face pose estimation.

A. Facial Animation Parameters

Facial Animation Parameters (FAPs) [1] are a set of parameters defined by MPEG-4 for the animation of synthetic face models. There are 68 FAPs (2 high-level FAPs for visual phoneme and expression and 66 low-level FAPs). The 66 low-level FAPs are used to characterize the facial feature movement over jaw, lips, eyes, mouth, nose, cheek, ears, etc. The FAPs are intended to be exhaustive, however, not all the FAPs are active in face expressions. Thus, we select 27 FAPs associated with facial expressions which are adequate to recognize the six basic facial expressions, as summarized in Table I. The FAPs are computed through tracking a set of facial features defined in Fig. 4. FAPs are measured by facial animation parameter units (FAPUs) that permit us to place the FAPs on any facial model in a consistent way. The FAPUs are defined with respect to the distances between key facial features in their neutral state such as eyes (ES0), eyelids (IRDS0), eye-nose (ENS0), mouth-nose (MNS0), and lip corners (MW0), as shown in Fig. 4. The facial feature points in Fig. 4 provide spatial references for defining FAPs. Table II gives a list of FAPUs. Notice that, the feature points 5.3 and 5.4 for cheek raising (see Fig. 4) are not tracked due to its unreliability in tracking, but these FAPs can be inferred in FAP reconstruction.

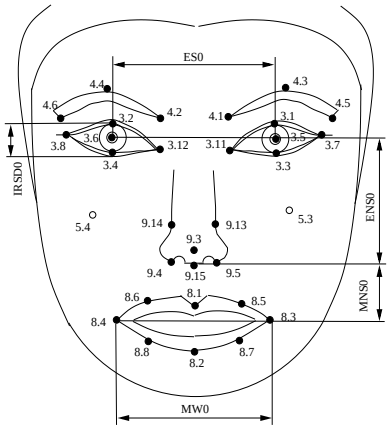


Fig. 4. A neutral face model and feature points used to define facial animation parameter units (FAPU). The feature points are numerated with MPEG-4 visual standard. Only the feature points marked with solid dots are tracked.

TABLE I

FACIAL ANIMATION PARAMETERS ASSOCIATED WITH THE SIX FACIAL EXPRESSIONS

Group	Facial Animation Parameter (FAP)	Expressions
2	open_jaw, raise_b_midlip, stretch_r_cornerlip, raise_l_cornerlip, raise_r_cornerlip, push_b_lip, stretch_l_cornerlip, depress_chin	Happiness, Surprise, Anger,Sadness Disgust
3	close_t_l_eyelid, close_t_r_eyelid close_b_l_eyelid, close_b_r_eyelid	Happiness, Sadness, Anger,Fear, Surprise
4	raise_l_i_eyebrow, raise_r_i_eyebrow, raise_l_o_eyebrow, raise_r_o_eyebrow, squeeze_l_eyebrow, squeeze_r_eyebrow	Anger, Happiness, Fear
5	lift_l_cheek, lift_r_cheek	Happiness
8	raise_b_midlip_o, stretch_l_cornerlip_o, stretch_r_cornerlip_o, raise_l_cornerlip_o, raise_r_cornerlip_o	Happiness, Sadness, Anger
9	stretch_l_nose, stretch_r_nose	Disgust,Anger

TABLE II

FACIAL ANIMATION PARAMETER UNIT (FAPU)

FAPU	Measurement	Description	FAPU value
IRISD0	$D_y(3.1-3.3)$ $D_y(3.2-3.4)$	IRIS diameter	IRISD = IRISD0/1024
ES0	$D_x(3.5-3.6)$	Eye Separation	ES = ES0 / 1024
ENS0	$D_y(3.5-9.15)$	Eye-Nose Separation	ENS = ENS0 / 1024
MNS0	$D_y(9.15-2.2)$	Mouth-Nose Separation	MNS = MNS0 / 1024
MW0	$D_x(8.3-8.4)$	Mouth width	MW = MW0 / 1024
AU		Angular Unit	10^{-5} rad

B. Facial Feature Detection

To extract FAPs, facial feature points have to be detected since they provide spatial reference for defining FAPs. Our technique in facial feature detection starts with face detection and then eye detection on the detected face using approach described in [37]. Both face and eye detector use an AdaBoost algorithm [38] with geometric Haar features. The set of Haar features comprises of over 50,000 individual features, which is significantly larger than the actual amount of pixels in a training image (120x120 pixels for a face training image and 30x30 pixels for a eye training image in our case). The feature selection and classifier construction are done simultaneously by using AdaBoost. In AdaBoost, each training sample is initialized with an equal weight, and weak classifiers are constructed using the individual Haar features.

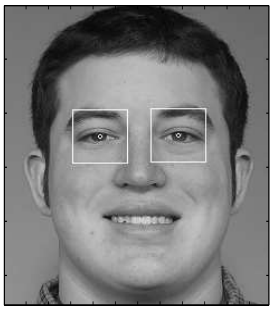


Fig. 5. An example of detected eyes and pupils (marked with white circles) by using the Adaboost classifier and Haar features.

The samples are then re-weighted based on the performance of the selected weak classifier, and the process is repeated. AdaBoost forces the classifier to focus on the most difficult samples in the training set, and thus it results in a very efficient classifier. To speed up the algorithm, the final classifier is broken up into a series of cascaded AdaBoost classifiers. Consequently, a large majority of the negative samples (non-face or non-eyes) are removed in earlier cascades and this results in a much faster real time classifier. The final classifier was trained with over 40,000 positive samples (face or eye) and hundreds of thousands of negative samples. The face and eye samples are cropped from numerous DVD movies. The final classifier utilizes nearly 1000 Haar features, with a final positive rate of nearly 99 percentage, and a final false positive rate of 10^{-7} percentage. Fig. 5 shows an example of detected eyes and pupil positions.

Given the detected eyes, the image is first normalized and the normalized image is then used to detect other facial features. Each feature point $\mathbf{x}$ and its local neighborhood surrounding $\mathbf{x}$ are represented by a set of multi-scale and multi-orientation Gabor wavelet coefficients. The Gabor kernel $\psi_{\mathbf{k}}(\mathbf{x})$ can be formulated as

$$\psi_{\mathbf{k}}(\mathbf{x}) = \frac{k^2}{\sigma^2} exp(-\frac{k^2 x^2}{2\sigma^2}) \left[exp(i\mathbf{kx}) - exp(-\frac{\sigma^2}{2}) \right], \quad (1)$$

where $\mathbf{k}$ is the characteristic wave vector, i.e., $\mathbf{k} = [k_i \cos\phi, k_i \sin\phi]^T$. We use $\sigma = \pi$, three spatial frequencies with wave numbers $k_i \in \{\frac{\pi}{2}, \frac{\pi}{4}, \frac{\pi}{8}\}$, and 6 orientations ($\phi$) from 0 to π differing by $\frac{\pi}{6}$. For each feature point, we compute a set of 18 complex Gabor wavelet coefficients. At each frame, the initial positions of each facial feature are located via Gabor wavelet matching in the approximate region constrained by the detected eyes. To achieve a robust and accurate detection, the initial feature positions are further refined by an active shape model that characterizes the spatial relationships between the detected facial features. Details about this work may be found in [39], [40].

C. Face Pose Estimation

The objective of face pose estimation has two-folds: 1) the face pose may distort the FAPs if they are computed directly from the 2D images, which has to be eliminated in the facial expression analysis; 2)

the face pose needs to be animated in order to generate a realistic facial expression on a synthetic face model.

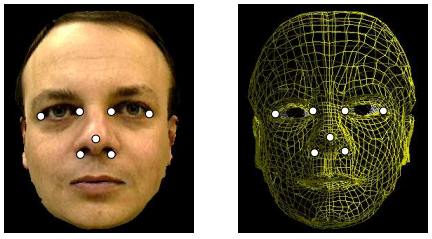
To estimate the face pose, we use a 3D face shape model (3D-FSM) and 7 relatively rigid (or near rigid) facial features under facial expressions including four eye corners and three points on the nose as control points to determine the 3D head movement, as shown in Fig. 6. Let (u_i, v_i) and (x_i, y_i, z_i) be the coordinates of the facial feature point i in 2D and 3D respectively. The coordinate (x_i, y_i, z_i) of a facial feature point i in the 3D-FSM is initialized by using a generic 3D-FSM. After the positions of facial features are detected from a frontal neutral face, the 3D-FSM is calibrated by those detected 2D feature positions, but its depth is preserved. Now we want to estimate the face pose, i.e., pan, tilt and swing angles (ω, ϕ, κ) and a scale factor (λ) related to the distance between the face and the camera. To this end, we assume that the face is not very close to the camera, then we can use the weak perspective model defined by the matrix M for an adequate approximation:

$$M = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \end{pmatrix}. \tag{2}$$

The two vectors $\mathbf{m_1} = (m_{11}\ m_{12}\ m_{13})$ and $\mathbf{m_2} = (m_{21}\ m_{22}\ m_{23})$ represent the first and second row of the rotation matrix, multiplied by the λ such that they are equal in length and orthogonal, i.e., $\mathbf{m_1} \cdot \mathbf{m_1} = \mathbf{m_2} \cdot \mathbf{m_2}$ and $\mathbf{m_1} \cdot \mathbf{m_2} = 0$. Then, each facial feature (x_i, y_i, z_i) in the 3D-FSM and its corresponding 2D image point (u_i, v_i) are related as follows:

$$\begin{pmatrix} u_i - u_0 \\ v_i - v_0 \end{pmatrix} = M \begin{pmatrix} x_i - x_0 \\ y_i - y_0 \\ z_i - z_0 \end{pmatrix}, \tag{3}$$

where $i = 1 \cdots 7$ are the control points as given in Fig. 6, and $(u_0, v_0)^T$ and $(x_0, y_0, z_0)^T$ are the centroids of the 7 points in 2D and 3D respectively. M can be estimated in least-square senses using the 7 points. Given M, the 3D pose (ω, ϕ, κ), a 2D translation, and scale factor (λ) can then be computed.



(a) (b)

Fig. 6. The 3D facial shape model: (a) a frontal face image; (b) the 3D face shape model with 7 rigid facial features (marked with white dot) as control points.

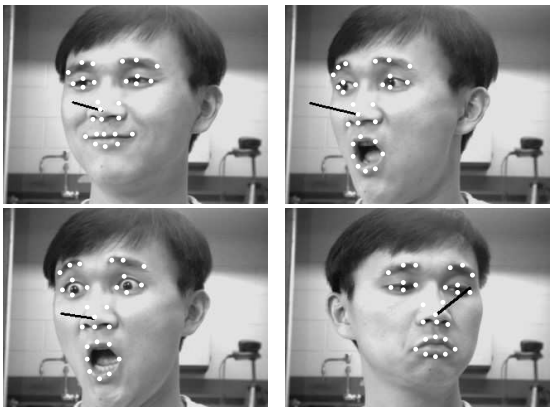


Fig. 7. An illustration of facial feature and pose tracking under facial expressions. Here the face normal is represented by a dark line and the detected features are marked with white dots.

In summary, the use of the facial shape model [41] combined with eye detection technique [37] makes our facial tracking and pose estimation very robust under large rigid head motion and non-rigid facial expression. Specifically, The allowed out-of-plane head rotation is around $\pm 40^{\circ}$. Fig. 7 illustrates a facial tracking example, where the face normal perpendicular to the face plane is computed from the three estimated Euler face pose angles.

D. FAP Analysis

Once the face pose matrix M is estimated, the 3D coordinate of any facial feature can be recovered by eliminating the face pose effect from its 2D coordinate in the image. Specifically, since the z coordinate value of each 3D facial feature is adapted from a neutral generic 3D face model directly, Eq. (3) can be rewritten as follows:

$$\begin{pmatrix} x_i - x_0 \\ y_i - y_0 \end{pmatrix} = M_{2\times 2}^{-1} \begin{pmatrix} u_i - u_0 \\ v_i - v_0 \end{pmatrix} - M_{2\times 2}^{-1} \begin{pmatrix} m_{13} \\ m_{23} \end{pmatrix} \times (z_i - z_0), \tag{4}$$

where

$$M_{2\times 2} = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix}. \tag{5}$$

Therefore, via Eq. (4), the 3D coordinate (x_i, y_i, z_i) of each facial feature can be obtained once the face pose matrix M is estimated. Subsequently, based on the recovered 3D coordinate of each facial feature, the associated FAPs can be computed directly. Table III exploits the translations from facial features to FAPs. Notice that the relation between FAPs and AUs in this table will be discussed in Section V. The

TABLE III

THE RELATIONSHIP BETWEEN FAPS AND AUS, AND FAP MEASUREMENT WITH FACIAL FEATURE POINTS

FAP Number	FAP Name	Distance of Two Feature Points	FAPU	AU
31	raise_l_i_eyebrow	$D_y(4.2, 3.8)$[3]	ENS	AU1
32	raise_r_i_eyebrow	$D_y(4.1, 3.11)$	ENS	
35	raise_l_o_eyebrow	$D_y(4.6, 3.12)$	ENS	AU2
36	raise_r_o_eyebrow	$D_y(4.5, 3.7)$	ENS	
31_	raise_l_i_eyebrow [1]	$D_y(4.2, 3.8)$	ENS	AU4
32_	raise_r_i_eyebrow	$D_y(4.1, 3.11)$	ENS	
37	squeeze_l_eyebrow	$D_x(4.4, 3.8)$	ES	
38	squeeze_r_eyebrow	$D_x(4.3, 3.11)$	ES	
19_	open_t_l_eyelid	$D_y(3.6, 3.2)$	IRSD	AU5
20_	open_t_r_eyelid	$D_y(3.5, 3.1)$	IRSD	
19	close_t_l_eyelid	$D_y(3.6, 3.2)$	IRSD	AU6
20	close_t_r_eyelid	$D_y(3.5, 3.1)$	IRSD	
41	lift_l_cheek [2]	$D_y(5.4, 3.12)$	ENS	
42	lift_r_cheek	$D_y(5.3, 3.11)$	ENS	
21	close_b_l_eyelid	$D_y(3.4, 3.6)$	IRSD	AU7
22	close_b_r_eyelid	$D_y(3.3, 3.5)$	IRSD	
61	stretch_l_nose	$D_y(9.14, 3.8)$	ENS	AU9
62	stretch_r_nose	$D_y(9.13, 3.11)$	ENS	
59	raise_l_cornerlip_o or	$D_y(8.4, 3.12)$	MNS	AU10
60	raise_r_cornerlip_o	$D_y(8.3, 3.11)$	MNS	
59	raise_l_cornerlip_o	$D_y(8.4, 3.12)$	MNS	AU12
60	raise_r_cornerlip_o	$D_y(8.4, 3.11)$	MNS	
53	stretch_l_cornerlip_o	$D_x(8.4, 9.15)$	MW	
54	stretch_r_cornerlip_o	$D_x(8.3, 9.15)$	MW	
59_	lower_l_cornerlip	$D_y(8.4, 9.15)$	MNS	AU15
60_	lower_r_cornerlip	$D_y(8.3, 9.15)$	MNS	
5	raise_b_midlip	$D_y(8.2, 9.15)$	MNS	AU16
16	push_b_lip	$D_y(8.2, 8.1)$	MNS	
18	depress_chin	$D_y(8.2, 9.15)$	MNS	AU17
53	stretch_l_cornerlip	$D_x(8.4, 8.3)$	MW	AU20
54	stretch_r_cornerlip	$D_x(8.3, 8.4)$	MW	
5	raise_b_midlip	$D_y(8.2, 9.15)$	MNS	
53_	tight_l_cornerlip	$D_x(8.4, 8.3)$	MW	AU23
54_	tight_r_cornerlip	$D_x(8.3, 8.4)$	MW	
4	lower_t_midlip	$D_y(8.1, 9.15)$	MNS	AU24
16	push_b_lip	$D_y(8.2, 9.15)$	MNS	
17	push_t_lip	$D_y(8.1, 9.15)$	MNS	
3	open_jaw (slight)	$D_y(8.2, 8.1)$	MNS	AU25
5_	lower_b_midlip (slight)	$D_y(8.2, 9.15)$	MNS	
3	open_jaw (middle)	$D_y(8.2, 8.1)$	MNS	AU26
5_	lower_b_midlip (middle)	$D_y(8.2, 9.15)$	MNS	
3	open_jaw (large)	$D_y(8.2, 8.1)$	MNS	AU27
5_	lower_b_midlip (large)	$D_y(8.2, 9.15)$	MNS	

Note: 1. FAP$5_$, $31\_$, $32\_$, $53\_$, $54\_$, $59\_$, and $60\_$ denote the FAPs that their motion is in the opposite direction to FAP5, 31, 32, 53, 54, 59 and 60, respectively. We give our own name of these FAPs for convenience. 2. FAP41 and 42 are not used in our system since the feature points for these two FAPs are not tracked. 3. Facial feature points are numbered with MPEG-4 visual standard (see Fig. 4). $D_x(p_1, p_2)$ and $D_y(p_1, p_2)$ are respectively the distance of two points p_1 and p_2 in X and Y direction.

value of a FAP is represented by either D_x or D_y, where D_x or D_y is the distance of two 3D feature points $p_1(x_1, y_1, z_1)$ and $p_0(x_0, y_0, z_0)$ in the X and Y directions respectively, i.e., $D_x = |x_1 - x_0|$ and $D_y = |y_1 - y_0|$.

V. Facial Expression Analysis

In this section, we first give a descriptive model of facial expressions by using AUs and FAPs, and then a computational model is presented.

A. Facial Expressions with AUs

A facial expression is indeed a combination of AUs. The AUs relevant to the six facial expressions are given in Table IV. AUs can be grouped as primary AUs and auxiliary AUs for a specific facial expression [16]. By the primary AUs, we mean those AUs or AU combinations that can be clearly classified as or are strongly pertinent to one of the six facial expressions without ambiguity. In contrast, an auxiliary AU is the one that can only be combined with primary AUs as supplementary cue in distinguishing facial expressions. Therefore, a facial expression contains primary AUs and auxiliary AUs. For example, AU9 (Nose Wrinkler) can be directly associated with disgust, but it is ambiguous to associate AU17 (Chin Raiser) with disgust. When AU9 and AU17 appear simultaneously, the classification of this AU combination to disgust then becomes more certain. Thus, AU9 is a primary AU of disgust and AU17 is its auxiliary AU. Table V gives a summary of primary AUs and auxiliary AUs associated with the six facial expressions, which represents an extension to Ekman's work [2]. The above relations between AUs and facial expressions are captured by a probabilistic framework (see Section V-B) to model the relationships of AUs with emotional expressions in an analytical way.

To automatically quantify the activation of the muscles directly from a face image, we need quantitatively relate AUs to facial feature movements. FAPs in MPEG-4 visual standard provide a way in measuring the muscular actions that characterize a facial expression. FAPs can be used to quantitatively characterize the muscle movement specified by an AU. In other words, an AU can be coded by several FAPs. Table III gives the relations between FAPs and AUs.

B. Computational Model of Facial Expressions

Tables III and V deterministically characterize the relations between facial expressions and AUs and between AUs and FAPs. To account for the uncertainty in the feature measurement and the dependency among AUs, we cast the deterministic relations into a probabilistic framework using a Bayesian network (BN) [42]. The BN model provides us a mathematically rigorous foundation for consistent, coherent and efficient reasoning and visual information fusion.

TABLE IV

A LIST OF AUS RELEVANT TO THE SIX FACIAL EXPRESSIONS

AU	Description	AU	Description
AU1	Inner brow raiser	AU2	Outbrow raiser
AU4	Brow Lower	AU5	Upper lid raiser
AU6	Cheek raiser	AU7	Lid tighter
AU9	Nose wrinkler	AU10	Upper lip raiser
AU12	Lip corner puller	AU15	Lip corner depressor
AU16	Lower lip depressor	AU17	Chin raiser
AU20	Lip stretcher	AU23	Lip tighter
AU24	Lip pressor	AU25	Lip apart
AU26	Jaw drop	AU27	Mouth stretch

TABLE V

AUS CLASSIFICATION VIA FACIAL EXPRESSIONS

Expressions	Primary AUs	Auxiliary AUs
Happiness	6, 12	25, 26, 16
Sadness	1, 15, 17	4, 7, 25, 26
Disgust	9, 10	17, 25, 26
Surprise	5, 26, 27, 1+2	
Anger	2, 4, 7, 23, 24	17, 25, 26, 16
Fear	20, 1+5, 5+7	4, 5, 7, 25, 26

Note: $i + j$ in the table indicates the combination of AUi and AUj.

Based on the relationship between a facial expression and a group of AUs, and between an AU and a group of FAPs as in Table V and Table III, the BN model of facial expressions may have three different abstractions: expression layer, facial AU layer and FAP layer as shown in Fig. 8. The expression layer consists of the root node, and a set of attribute variables denoted as HAP, ANG, SAD, DIS, SUP and FEA corresponding to the six facial expressions. We assume that an image sequence only contains the six facial expressions plus a neutral state. If the probability of the six facial expressions are equally distributed, the face is neutral. The emotional intensity is measured by the probability distribution over the six facial expressions on the top node. The AU layer captures the relation between AUs and facial expressions as given in Table V. A primary AU contributes stronger visual cue to the understanding of the facial expression than an auxiliary AU does. This is quantified by their conditional probabilities. FAPs occupy the lowest level of layers and they are only observable variables in the model. A FAP amplitude is discretized into multiple levels to differentiate the intensity of a muscular action. Considering the measurement accuracy and the complexity of conditional probability table, we use 3 amplitude levels (low, middle, high) for each FAP, where the values are determined by statistically analyzing Cohn-Kanade

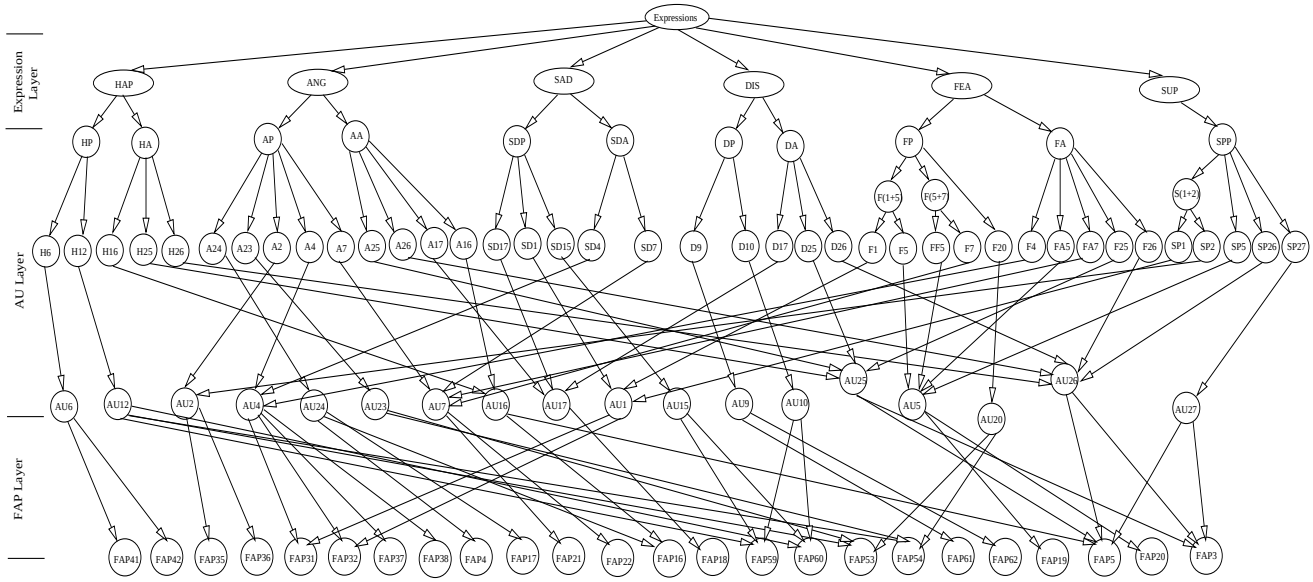


Fig. 8. The BN model of the six facial expressions. The notations HAP, ANG, SAD, DIS, FEA and SUP denote Happiness, Anger, Sadness, Disgust, Fear and Surprise, respectively. Here are notational examples. HP, AP, SDP, DP, FP and SPP denote the primary AUs of happiness, anger, sadness, disgust, fear and surprise, respectively. HA, AA, SDA, DA and FA denote the auxiliary AUs of happiness, anger, sadness, disgust and fear, respectively. H_i, A_i, SD_i, D_i, F_i and S_i denote AU_i belonging to happiness, anger, sadness, disgust, fear and surprise, respectively. $F(1+5)$ denotes the combination of AU1 and AU5, which belongs to fear. FAPi is a FAP with number i. For clarity, the nodes FAP5_, 31_, 32_, 53_, 54_, 59_, and 60_ are represented by FAP5, 31, 32, 53, 54, 59 and 60, respectively, and they are represented by separate nodes in our implementation.

facial expression database [43]. Notice that the intensity of FAPs does not represent the intensity of the intended facial expression. The emotional intensity for each expression is measured by its probability. The conditional probabilities are estimated by Maximum Likelihood Estimation by using FAPs extracted from 200 sequences of 50 subjects covering the 6 expressions in Cohn-Kanade facial expression database [43]. Then, we manually tune the learned conditional probabilities in order to achieve the best result in facial expression analysis. For the training data, the facial feature points are manually detected for the accuracy reason.

Facial expressions can be said to express emotions which vary according to subject-environment interaction. A facial expression starts with the muscular contraction and its intensity increases until the apex reaches. Then it gradually releases to the neutral. Thus, a facial expression often reveals not only the nature of the deformation of facial features, but also the relative timing of facial actions and their temporal evolution. Modeling such a temporal course allows us synthesize the emotional evolvement so that we can generate the dynamic effects in the animation of human expressions.

Dynamic Bayesian networks (DBNs) were proposed as a generalization of Hidden Markov Model (HMM). It is natural to consider DBNs as a basis of the general spatio-temporal sensor data analysis and interpretation since DBNs allow much more general graph structures than HMMs. Our DBN model of

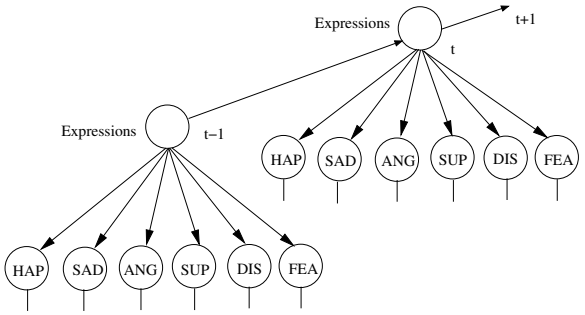


Fig. 9. The temporal links of DBN for modeling facial expression (only two time slices are shown since the structure repeats by "unrolling" the two-slice BN). Node notations are given in Fig. 8.

facial expressions is made up of interconnected time slices of a static BN, and the dependency between two neighboring time slices are based on first order HMM. The relative timing of facial actions during the emotional evolution is expressed by moving a time frame in accordance with the frame motion of a video sequence so that the visual information at the previous time provides diagnostic support for current hypothesis. Fig. 9 shows the temporal dependencies by linking the top nodes of the BN given in Fig. 8. Since the preceding evidences is completely summarized in the hypothesis at previous time slice, the belief of the current hypothesis is inferred relying on the combined information of current visual cues through causal dependencies in the current time slice, as well as the preceding evidences through temporal dependencies. Consequently, the probability of facial expressions from the preceding time slice serves as a prior information for current hypothesis. The prior information is integrated with current visual evidences to produce a posterior estimate of current facial expression. Let Θ be a hypothesis variable of facial expression, $S^{(1)}, \cdots, S^{(n)}$ be the n intermediate variables in the DBN, and $\mathbf{e}$ be a set of visual observations. The probability that we are interested in is the posterior distribution of the six facial expressions given a set of visual observations (FAPs), i.e., $P(\Theta_t|\mathbf{e})$. Applying Bayes' theorem, we have

$$P(\Theta_t|\mathbf{e}) = \frac{P(\Theta_t, \mathbf{e})}{\sum_{\Theta_t} P(\Theta_t, \mathbf{e})}, \tag{6}$$

where t is the discrete time index, and

$$\begin{aligned} P(\Theta_t, \mathbf{e}) = \sum_{\Theta_{t-1}} \sum_{S_t^{(i)}} \Big\{ & P(\Theta_t|\Theta_{t-1}) P(\Theta_{t-1}) \\ & \times \prod_i P(S_t^{(i)}|\pi(S_t^{(i)}) P(\mathbf{e}|\pi(\mathbf{e})) \Big\}, i = 1, \cdots, n, \end{aligned} \tag{7}$$

where $\pi(*)$ denotes the parents of node $*$, and Θ_t and Θ_{t-1} are the hypothesis at time t and $t-1$, respectively. $P(\Theta_t|\Theta_{t-1})$ in Eq. (7) is the state transition probability of the hypothesis node between two consecutive time slices; $P(\Theta_{t-1})$ is the prior probability of hypothesis at the current time and the posterior probability of hypothesis at the preceding time. The above equations can be solved for by using

efficient DBN inference algorithms. The details of Bayesian inference algorithm are beyond the scope of this paper. A thorough work on DBN inference and learning can be found in [44].

The current state of a facial expression is inferred by fusing systematically the current FAPs and the previous FAPs. If the same evidence is acquired sequentially, the evidence acquired at current time can reinforce the hypothesis made by the same information received at previous time. DBNs enable to correlate and associate the continual arriving evidences through temporal dependencies to perform reasoning over time. The information from previous time serves as prior information for current evidences, and they are combined with Bayesian statistics. As indicated in [4], information about the temporal course of a facial action may have psychological meaning relevant to the intensity. This means that the current state of a facial expression and its intensity shall be reasoned over time. The DBN model integrates visual evidences over time to reason about a facial expression, which is less influenced by the missed FAP extraction at current time. This is directly beneficial to the perceptual quality of the resulting animation.

VI. Facial Expression Synthesis

For the state-of-the-art solutions with MPEG-4 visual standard, to reproduce the facial expressions on the client-side, FAPs have to be transmitted to the synthesizer. To accommodate very low bandwidth constraint, the FAPs must be compressed. There are several shortcomings. First, FAP missed extraction will create strange animation artifacts which directly affect the perceptual quality of the resulting animation. Second, some FAPs such as FAP41 and FAP42, which are unmeasurable by automatic video analyzer due to the difficulty in detecting facial feature point 5.3 and 5.4 on the cheek (see Fig. 4), also affect the perceptual quality of the resulting animation. Third, for very low bit-rate and interactive applications, PCA technique is used to achieve efficient FAP compression for intraframe coding with the loss of reconstruction accuracy. Our approach described as follows overcomes these problems.

At the synthesizer, we use a static BN that has the same spatial structure as the DBN model in facial expression analysis. Two BNs are coupled to unify the facial expression analysis and synthesis into one coherent structure so that the visual evidences observed at the analysis end can be propagated directly to the synthesizer for reconstructing the FAPs and their intensity. Fig. 10 depicts the dependency graph of such a coupled BN. The BN model is coupled with the DBN at analysis end by the conditional dependency between their top nodes Θ and Θ' so that the probability distribution of six facial expressions at analysis end passes to the BN at the synthesizer, which would be performed exactly as data stream channel in the actual applications. At the analysis end, the hypothesis node Θ (the probability distribution of the six facial expression) completely summarizes the continual arriving visual evidences (FAPs) by integrating them with Bayesian statistics. At the synthesizer, the FAPs are inferred given Θ'. Because Θ and Θ' are conditionally dependent, the visual evidences (FAPs) at the facial expression analysis end directly

propagates to the synthesizer to reason about the FAPs and their intensity. The dependency link between Θ and Θ' can be viewed as data stream channel between the two ends. Therefore, in the actual applications, we only need to transmit the state of Θ (the probability distribution of the six facial expressions) to the synthesizer for reconstructing the FAPs. Since all data being transmitted can be represented by using 8 bits in sufficient precision, we need to transmit 6 bytes of data per frame for reconstructing the FAPs plus 3 bytes for 3 angles of face pose information (each angle is less than $90°$).

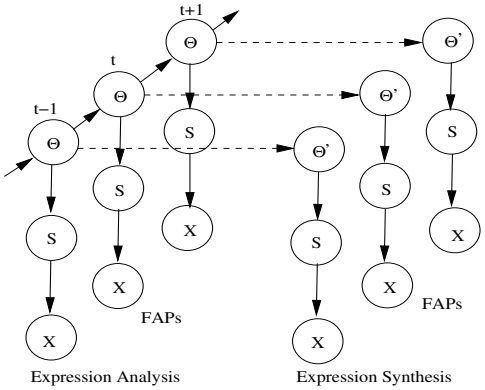


Fig. 10. A dependency graph of a coupled Bayesian network. The facial expression analysis end uses a dynamic Bayesian network and the facial expression synthesizer uses a static Bayesian network (see Fig. 8 for the detail of BN model). The terms Θ and Θ', S, and X denote the hypothesis nodes, a set of intermediate nodes and a set of FAPs, respectively.

In terms of Bayesian networks, we usually have visual evidences of an effect to infer the most likely cause. This is called diagnostic, or "bottom up", reasoning, since it goes from effects to causes as can be seen in Eq. (6). Bayesian networks can also be used for causal, or "top down", reasoning to specify how causes generate effects. Hence, we can compute the probability that a set of FAPs are generated given the state of Θ', where the state of Θ' is equal to the state of Θ if Θ' and Θ are 100% dependent. Now let X_j be a specific FAP, $S = \{S_1, \cdots, S_n\}$ be a set of intermediate nodes interconnecting Θ' and X_j. Given a set of visual readings $\mathbf{e}$ at the analysis end, the probability of a specific FAP X_j that can be generated at the synthesis end may be written as

$$P(X_j|\mathbf{e}) = \sum_{\Theta', S_i} \Big\{ P(X_j|\pi(X_j)) \prod_i P(S_i|\pi(S_i)) \times P(\Theta'|\mathbf{e}) \Big\}, i = 1, \cdots, n, \tag{8}$$

where $\pi(*)$ denotes the parents of node $*$. Since

$$P(\Theta'|\mathbf{e}) = \sum_{\Theta} P(\Theta'|\Theta) P(\Theta|\mathbf{e}), \tag{9}$$

then we have

$$P(X_j|\mathbf{e}) = \sum_{\Theta,\Theta',S_i} \Big\{ P(X_j|\pi(X_j)) \prod_i P(S_i|\pi(S_i)) \times P(\Theta'|\Theta) P(\Theta|\mathbf{e}) \Big\}, i = 1, \cdots, n, \tag{10}$$

where $P(\Theta|\mathbf{e})$ received from the analysis end is updated when new visual readings input to the DBN model. Here $P(X_j|\mathbf{e})$ provides quantitative information about the evolving facial features. Because $P(X_j|\mathbf{e})$ is a probability distribution, FAP nodes in the BN model only need binary state (true or false) so that we can use $P(X_j = true|\mathbf{e})$ to reconstruct a FAP and its intensity scale. Again, the conditional probabilities of the BN model are learned from facial expression database. We can see from Eq. (10) that the FAP intensity evolves as a function of $P(\Theta|\mathbf{e})$, and $P(\Theta|\mathbf{e})$ is the result that combines the continual arriving visual evidences over time with Bayesian statistics. At the synthesis end, the FAPs that are the most relevant to the current state of facial expressions have higher probability than others. In other words, the FAP intensity at the synthesis end evolves based on the continual arriving visual evidences (FAPs) at the analysis end.

There are several benefits with this approach for FAP reconstruction:

1) Since the conditional probabilities in the model are parameterized by learning from facial expression databases, all FAPs relevant to a facial expression can be inferred. Thus, a FAP that may be not extracted by video analyzer or the unmeasurable FAPs such as FAP41 and FAP42 can also be reasoned at the synthesis end. This improves the perceptual quality of the resulting animation.
2) To reconstruct the FAPs, the synthesizer needs only the probability distribution of six facial expressions $P(\Theta|\mathbf{e})$ so that a very low bit-rate (9 bytes per frame) in data transmission can be achieved.
3) Unlike the direct transmission of a stream of FAPs to the synthesizer, our approach will not generate strange animation artifacts by the FAP extraction errors, e.g., a FAP that may not be extracted for several frames. This is because the intensity of all FAPs in the synthesizer always varies simultaneously according to the given $P(\Theta|\mathbf{e})$ though the extraction error affects $P(\Theta|\mathbf{e})$.

Now let f be FAP amplitude value and f_{max} be its maximal amplitude value. Let p_n, p_c and p_a be the intensity scale of this FAP when a facial expression in neutral state, current state and apex state, respectively. Then, the FAP amplitude value that quantifies the movement of each FAP involved in the facial expression can be simply interpreted as

$$f = \frac{p_c - p_n}{p_a - p_n} f_{max}. \tag{11}$$

The value of f_{max} can be predetermined based on facial expression database in hand, and p_n, p_a can be obtained from the BN model. Table VI gives an example for happiness. Since f_{max} is measured by facial

animation parameter unit (FAPU), it allows us to map f on any facial model. To animate a pose-variable facial expression, we first position the feature point on the synthetic facial model, and the vertices around the feature point are deformed by interpolating them onto the predefined trajectory (a lookup table). This is performed only on the frontal view facial model. Then, the facial model is rotated according to 3D pose (ω, ϕ, κ) received from the analysis end.

TABLE VI

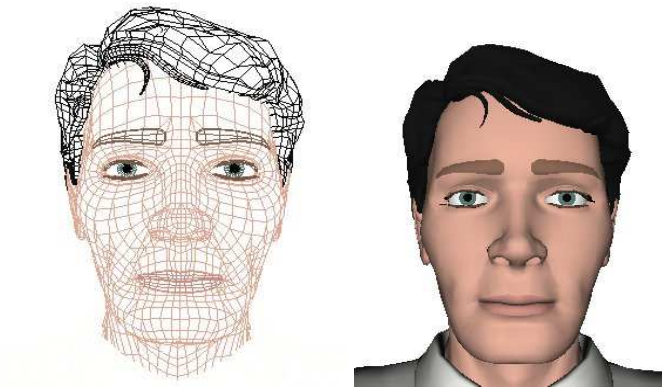
FAP VALUES OF HAPPINESS AT NEUTRAL AND APEX, AND THEIR MAXIMAL VALUES

FAPs	Neutral (p_n)	Apex (p_a)	Maximal Value (f_{max})
FAP3	0.1855	0.2730	213 (MNS0)
FAP19	0.2832	0.4860	101 (IRSD0)
FAP20	0.2832	0.4860	101 (IRSD0)
FAP41	0.1530	0.4780	122 (ENS0)
FAP42	0.1530	0.4780	122 (ENS0)
FAP53	0.2263	0.3295	173 (MW0)
FAP54	0.2263	0.3295	173 (MW0)
FAP59	0.1930	0.5400	316 (MNS0)
FAP60	0.1930	0.5400	316 (MNS0)

Given the amplitude of the FAPs, a facial expression can be reproduced on a synthetic facial model. Our 3D synthetic facial model consists of 3000 vertices and 3200 polygons to represent a generic face as shown in Fig. 11. The facial model allows rotation around its center given 3D Euler face pose angles. Our animation technique is not novel, and it is similar to the facial animation tables (FATs) in MPEG-4 visual standard. For each FAP, we define how the feature point has to move, i.e., the trajectory of feature point as a function of the amplitude of a FAP. After the motion of the feature points is defined for each FAP, we define how the motion of a feature point affects its neighboring vertices, i.e., the trajectory of neighboring vertices as a function of their feature point movement. By statistically analyzing facial expression database, we created lookup tables for mapping feature point motion onto vertex motion by specifying intervals of the FAP amplitude. Using the lookup tables, we interpolate the vertex movements by the linear approximation of vertex motion given FAP amplitude values so that deformable lattices can be generated on the facial model. Fig. 12 gives an example showing the deformation of the vertices around the mouth corners when a facial expression varies from its neutral state to the apex.

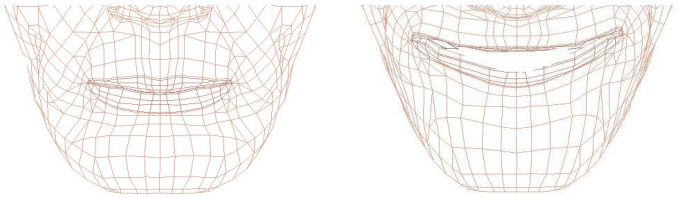
Our approach to the visual reproduction of facial expressions can be summarized as follows:

(1) Get visual observations (FAPs) $\mathbf{e} = \{e_1, \ldots, e_n\}$, and face pose (ω, ϕ, κ) by our video analyzer.
(2) Get the probability distribution of the six facial expressions $P(\Theta|\mathbf{e})$ by DBN inference given $\mathbf{e}$.
(3) Send $P(\Theta|\mathbf{e})$ and (ω, ϕ, κ) to the the synthesizer.
(4) Get the probability of FAPs $P(X_i = true|\mathbf{e})$ by BN inference given $P(\Theta|\mathbf{e})$.
(5) Obtain FAP amplitude value f_i based on $P(X_i = true|\mathbf{e})$.



(a) (b)

Fig. 11. (a) A wireframe facial model; (b) the synthetic facial model.



(a) (b)

Fig. 12. A wireframe of the mouth in neutral (a) and the apex of happiness (b), and the deformable vertices around the mouth. Given a FAP (here raising mouth corner), the deformation of vertices around the mouth corners is determined by the linear interpolation.

(6) Animate facial expression and orientation by mapping feature point motion onto vertex motion given a group of FAP amplitude values $\mathbf{f} = \{f_1, \ldots, f_n\}$ and face pose (ω, ϕ, κ).

VII. EXPERIMENTS

In this section, we first show how the dynamic nature of facial expression is modeled and then we perform the tests of facial expression synthesis and animation. Finally, the performance evaluation is given.

A. Facial Expression Analysis

A facial expression starts from a neutral state and its intensity increases gradually until the apex. After the maximum excursion of the muscle, its intensity decreases gradually until the neutral state. With regard to the intensity of the facial expression, the muscle contraction rates need to be measured at every stage of the emotional evolvement. However, since there are inter-personal variations with respect to the amplitudes of facial actions, it is practically difficult to determine the intensity of a given subject by machine extraction. The duration of an emotional expression is often related to the emotional intensity. This interprets that the current state of a facial expression can be inferred relying on the combined

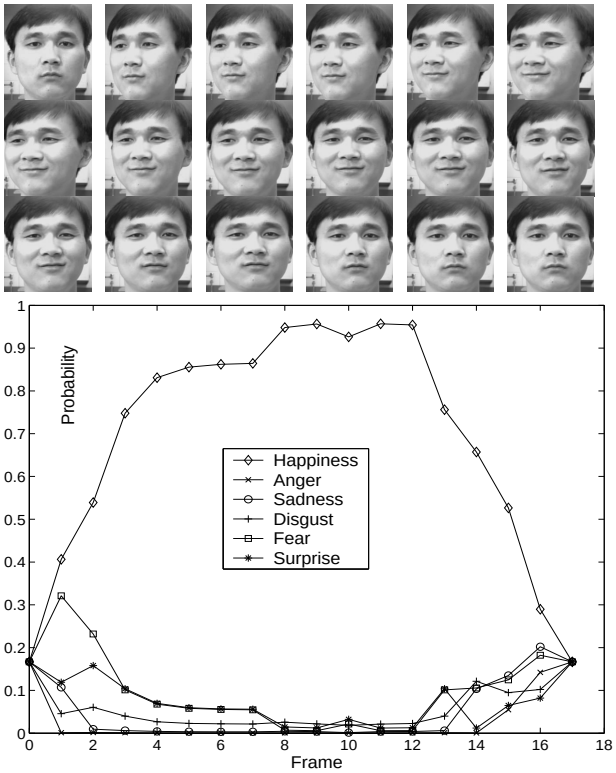


Fig. 13. Upper: a short image sequence showing that the subject starts with neutral, then its expression intensity reaches an apex and finally releases. Bottom: the probability distribution generated by our facial expression analysis model. The probability distribution here shows the temporal evolution of the facial expressions and it is indeed the intensity scale of the six facial expressions.

information of current visual cues as well as the preceding evidences. Hence, as we can observe from a typical result in Fig. 13, the evolution of emotional magnitude can be well modeled; this enables us to interpret the dynamics of the six basic emotional facial expressions. Another benefit of this approach can be best shown when the facial features are miss-detected. Our DBN facial expression model fuses systematically the current FAPs and the previous decision by Bayesian statistics. Therefore, even though the failure in extraction of some FAPs at current frame, the state of facial expression can still be inferred by reasoning over time. In addition, the presence of other facial features and their built-in relationships to the missing features in BN further help infer the facial expression. As shown in Fig. 14, this approach can well handle missing data.

Fig. 15 illustrates an output of this approach showing a temporal course of facial expressions resulted from sampling a video sequence in every 7 frames. The sequence has 700 frames sequentially containing the six facial expressions and the neutral states between two different expressions. Although, as we can see from the figure, there are a certain number of recognition errors (e.g., the confusions between surprise and fear) due to feature detection errors, the overall performance of modeling dynamics of emotional expressions appears good. In this paper, we are particularly interested in the capability of modeling the

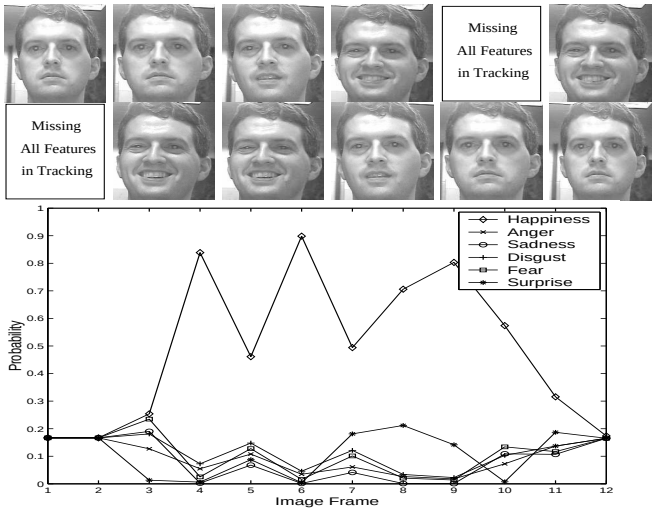


Fig. 14. Upper: An image sequence assuming that the facial features in some image frames are fully missed; Bottom: the intensity scale of the six facial expressions from our facial expression analysis model. The valleys of the curve at frame 5 and 7 are caused by the absence of certain facial features.

temporal course of a facial expression, rather than only the recognition accuracy for individual face images. The ability of our approach to correlate and reason about facial temporal information over time provides a coherent overview of the dynamic behavior of facial expressions in an image sequence so that various stages of the emotional evolvement can be analyzed by machine. This permits us to synthesize the realistic behavior of facial expressions with the analysis results.

B. Facial Expression Animation

First, we examine the FAP reconstruction quality. Assume we have the probability distribution of facial expressions received from the analysis end. For clarity, we only gives the intensity change of happiness as shown in Fig. 16. The figure depicts that the FAP intensity scale evolves as the intensity of happiness increases. We can see from the reconstructed FAPs that the FAPs relevant to happiness (FAP3, 19, 20, 41, 42, 53, 54, 59, 60) dominate other FAPs; this agrees with what is in Table V and Table III. In addition, FAP41 and FAP42 that are not measurable by our video analyzer due to that the feature points 5.3 and 5.4 for cheek raising are difficult to be detected, are inferred at the synthesis end due to their semantic relationships to other facial features.

Now we use the result from facial expression analysis as shown in Fig. 13 to illustrate the animation of facial expressions. Fig. 17 depicts the reconstructed FAP intensity curve according to the facial expression given in Fig. 13. It shows that the intensity scale of FAPs agrees with the evolution of the facial expression. Based on the FAPs provided in Fig. 17, we reproduce the temporal course of happiness on a synthetic facial model as shown in Fig. 18, where head rotation is also included. An additional results are given in

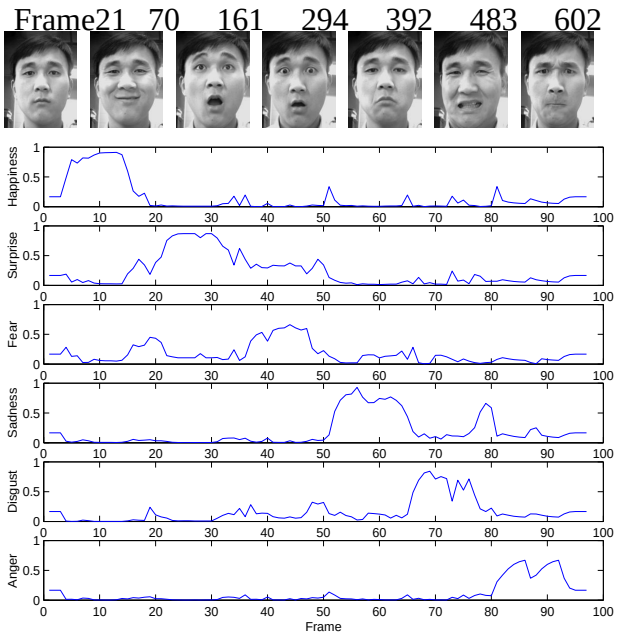


Fig. 15. Upper: a video sequence with 700 frames containing the six facial expressions and only 8 snapshots are shown for illustration. Bottom: the emotional intensity (probability distribution over the six facial expressions) plotted as a time series.

Fig. 19 which show that the temporal course of facial expressions can be well synthesized on a synthetic facial model with a reasonable visual quality.

However, this approach is unable to animate the personality of facial expressions. For example, the subject smiles with jaw closed in Fig. 13, while the jaw is slightly open in the synthesized result as can be seen in Fig. 18. In our facial expression model, the cause and effect relation between FAPs and facial expressions are determined uniquely by human physiology. In our case, we use Ekman's linguistic description of facial expressions. For happiness, when a subject performs smile, the mouth will slightly open and the degree of openness depends on the intensity of the expression. If a subject smiles without opening the jaw, equivalently, this is the case that FAP3 is not detected. However, the facial expression (here happiness) can still be determined by other relevant FAPs without FAP3. When reconstructing FAPs at the synthesis end, FAP3 is inferred due to its semantic relationships to other features. On the other hand, this is an interesting aspect of this approach, i.e., a missing FAP can be recovered at the synthesis end such that a missing FAP will not cause strange animation artifacts. One solution to personalize the animation result is to build a facial expression model for each individual; but it is practically less useful. As indicated by [4] that time course information is necessary for those desiring life-like facial animation because the time course of a facial action may have psychological meaning relevant to the intensity, genuineness, and other aspects of the expresser's state. In other words, our reconstruction is more faithful in facial expression than in FAPS. This is one of the main difference between our approach and other existing methods.

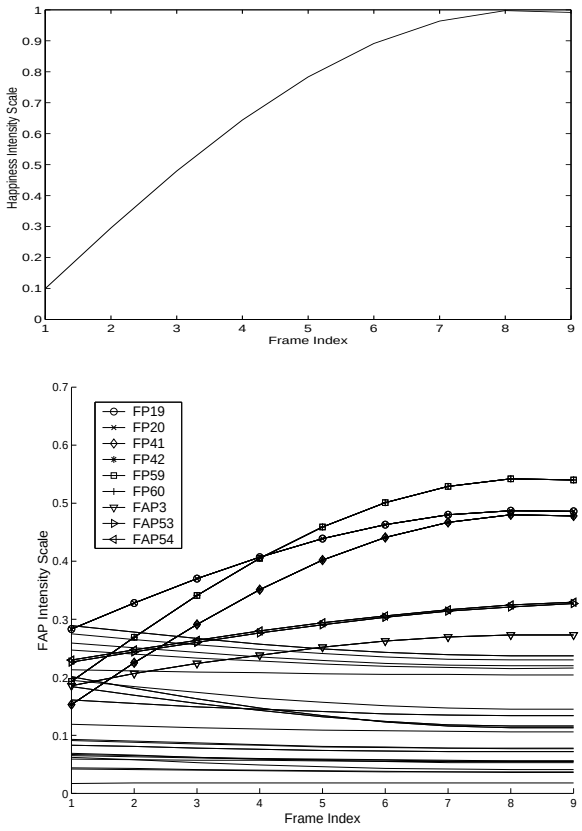


Fig. 16. Upper: assuming a perfect intensity curve of happiness varying from neutral to apex. Bottom: the intensity scale of FAPs evolves according to the intensity of the facial expression. The FAPs associated with this expression are marked with line symbols. Notice that, due to the symmetric nature of human face, the curves of FAP19, 41, 59, 53 overlap with the curves of FAP20, 42, 60, 54, respectively.

C. Performance Evaluation

In the literature, the peak signal-to-noise ratio (PSNR) between the original and the reconstructed FAP is often used for evaluating the FAP reconstruction quality. However, our approach is to model the temporal course of facial expressions by fusing FAPs over time based on Bayesian statistics. The FAPs are reconstructed by the BN inference given the distribution of the six facial expressions. We do not directly transmit the FAPs to the synthesizer. Therefore, the original and the reconstructed FAPs are expected to be different so that we cannot use PSNR to evaluate the reconstruction quality. Instead, we propose to quantitatively study the fidelity of facial expression reconstruction. Specifically, we perform the following experiments to demonstrate the performance of our system: 1) the dynamic effect of resulting facial animation, 2) the quality of facial expression reconstruction, and 3) the tolerance of the reconstructed facial expression to FAP measurement errors at the analysis end.

To observe how the FAPs evolve as a function of the received distribution of facial expressions, we manually create a perfect distribution of facial expressions, as shown in Fig. 20. The figure shows the temporal course of a facial expressions that a facial expression starts from a neutral state and its intensity

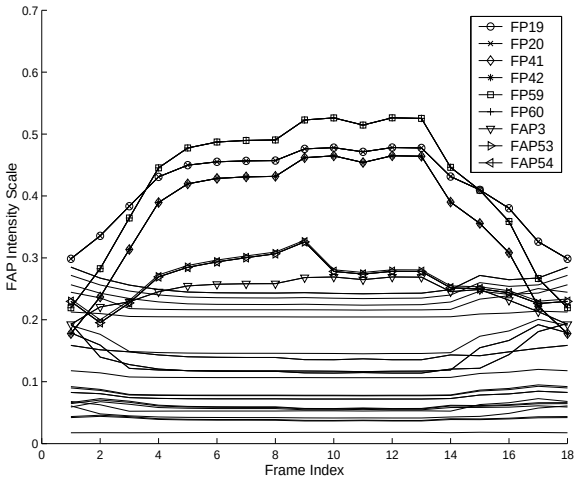


Fig. 17. The reconstructed FAP intensity scale evolves depending on the intensity of a facial expression. The FAPs associated with the facial expression are marked with line symbols. Notice that, due to the symmetric nature of human face, the curve of FAP19, 41, 59, 53 is overlapped by the curve of FAP20, 42, 60, 54, respectively. The result of facial expression analysis can be seen in Fig. 13.

Fig. 18. An example result for synthesizing the temporal course of happiness and head movement. The result of facial expression analysis can be seen in Fig. 13.

increases gradually until the apex and then releases gradually until the neutral state. Fig. 21 shows the reconstructed FAPs and their intensity. Two observations can be made from this result. First, the most significant reconstructed FAPs reflect current state of facial expressions. Second, the intensity scales of the reconstructed FAPs evolve smoothly in accordance with the intensity of facial expressions. The FAP intensity provides quantitative information about motions of the evolving facial expression. Therefore, the temporal course of facial expressions can be synthesized from the result of facial expression analysis. Fig. 22 presents the reconstructed FAPs using the result of facial expression analysis in Fig. 15, which is

Fig. 19. The animation result for the temporal course of sadness. The result of the facial expression analysis can be seen in Fig. 15 from frame 50 to 70.

indeed a noisy version of an ideal curve due to the detection errors from our automatic video analyzer.

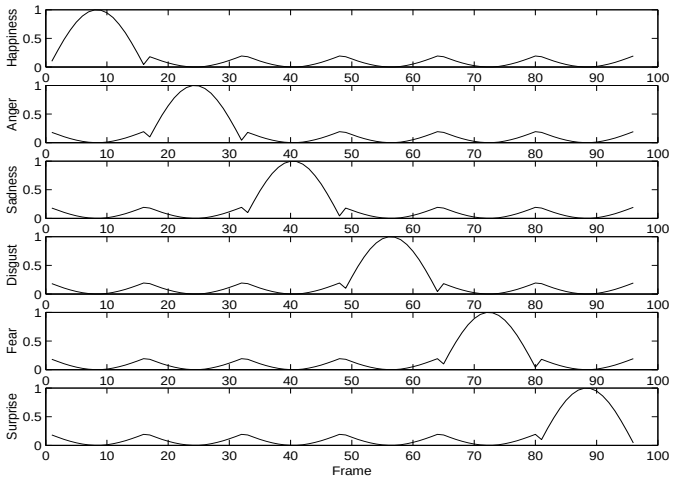


Fig. 20. A manually created intensity curve depicts an ideal temporal course of the six facial expressions. Assuming that the intensity of a facial expression follows an ideal curve starting from neutral to apex and returning to neutral.

To evaluate the fidelity of the reconstructed facial expression, we experimentally and quantitatively compare the reconstructed facial expression at the synthesis end with the original facial expression at the analysis end. The original expression at the analysis end is constructed using the detected FAPs at each frame while the reconstructed expression at the synthesis end is produced by the reconstructed FAPs. For this study, an image sequence from Cohn-Kanade facial expression database [43] is used. The sequence is "$S046\_005$" consisting of 23 frames and the subject performs a smile from the neutral state to the apex. Fig. 23 shows the comparison between two dynamic facial expressions, one based on the original detected FAPs at the analysis end and the other based on the reconstructed FAPs at the synthesis end. Notice that, for clarity, Fig. 23 plots only the probability of happiness (the probabilities of other five facial expressions are indeed negligible). Though the original FAPs and the reconstructed FAPs could be different, but we can see that the dynamic trends and the intensity of the two facial expressions are

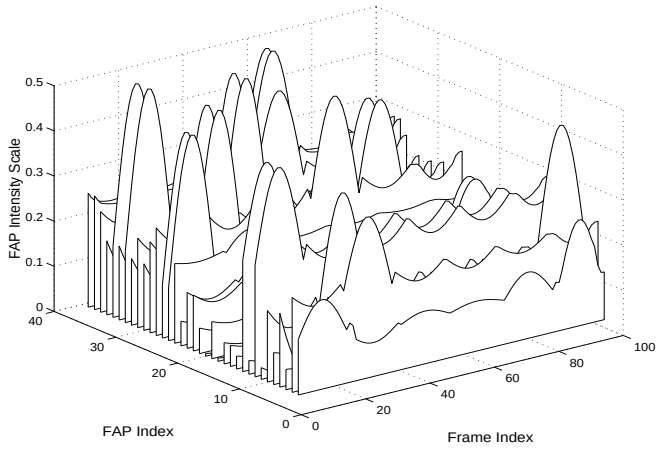


Fig. 21. FAPs are reconstructed at the synthesis end. It can be seen that the FAP intensity evolves depending on the intensity of facial expressions as given in Fig. 20.

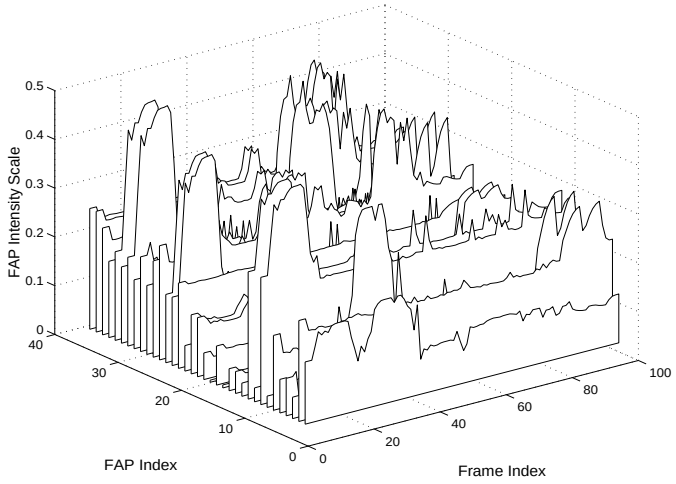


Fig. 22. FAPs are reconstructed at the synthesis end. This figure shows that the FAP intensity evolves depending on the intensity of facial expressions as given in Fig. 15. Due to the detection errors from our automatic video analyzer, the curves are not smooth.

very close. We tested a number of sequences in the database and the same results are obtained. This demonstrates our expression-faithful reconstruction of our method.

Finally, we use the same sequence as the above to test the tolerance of the reconstructed facial expression to FAP measurement errors at the analysis end. To simulate the measurement errors, we manually change FAP values. In this test case, FAP4 of frame 13 is changed to 0, and FAP4 of frame 15 is added by half of its value. Fig. 24 shows the comparison of resulting animation between our approach and the approach directly using the original FAPs. The result shows that our approach can tolerate the measurement errors without generating visible animation artifacts (the upper row of Fig. 24). On the other hand, if facial animation directly uses the original FAPs with FAP measurement errors from the facial expression analysis end, we can see that there are strange animation artifacts around the left corner of the mouth caused by the

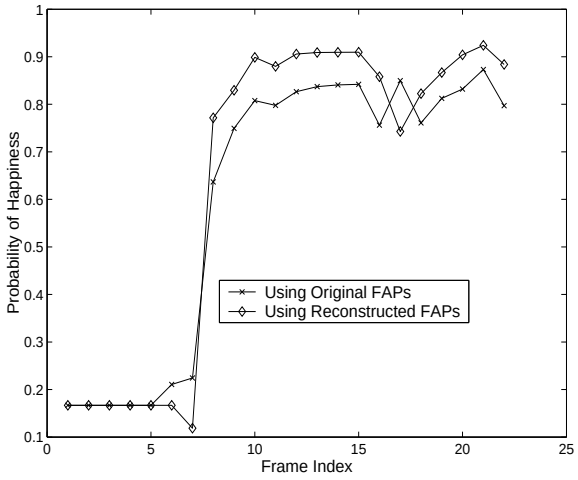


Fig. 23. A comparison between the facial expression generated by using the original FAPs and the facial expression generated by using the reconstructed FAPs via our facial synthesis method. Only the probability of happiness is shown.

FAP measurement errors (the bottom row of Fig. 24). In practice, an automatic video analyzer may often fail to detect feature points for various reasons such as image noise and light change. In our approach, the FAP measurement errors will not cause visible animation artifacts which affect the perceptual quality of the resulting animation.

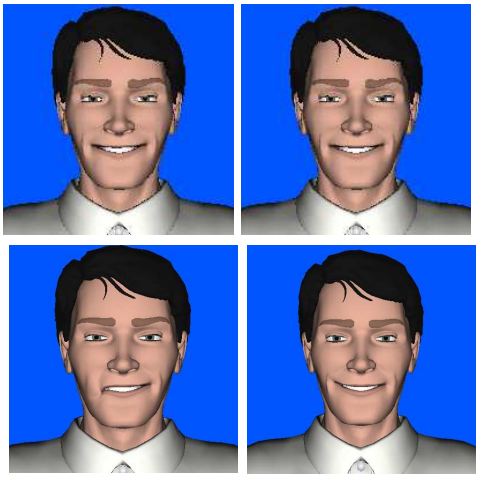


Fig. 24. The animation result of the 13th and 15th frame under FAP measurement errors (by manually changing FAP4 of the 13th to 0, and by adding FAP4 of the 15th frame the half of its value). Upper row: the animation result from our approach. It can be seen that the animation perceptual quality is not affected by some FAP measurement errors. Bottom row: the animation result by directly using the original FAPs from the facial expression analysis end. It can be seen that there are visible animation artifacts around the left corner of the mouth.

VIII. Conclusion

In light of MPEG-4 visual standard, a significant amount of research has been directed to MPEG-4 FAP compression and facial animation, but less emphasis has been placed on synthesizing the temporal course

of facial expressions since temporal course information is necessary for those desiring life-like facial animation [4]. This paper explores the use of a coupled Bayesian network to unify the facial expression analysis and synthesis into one coherent structure to synthesize dynamic facial expressions. Our approach enjoys the following major benefits:

1) To synthesize six pose-variable facial expressions, our approach needs to transmit 9 bytes of data per frame to the synthesizer. It is particularly suitable for the applications required a very low transmission rate to a remote synthesizer.
2) The temporal course of facial expressions can be synthesized by means of dynamically modeling the facial expression. This is particularly important for life-like facial animation.
3) Unlike the direct transmission of a stream of FAPs to the synthesizer, our approach would not generate visible animation artifacts if some FAPs are not extracted by automatic video analyzer. The perceptual quality of the resulting animation is less affected by FAP extraction errors.

However, like the some existing FAP compression methods that utilize the symmetric property of human face, our approach is incapable of reproducing personal facial expressions since 1) the semantic relations of FAPs and facial expressions are parameterized by linguistic descriptions of facial muscular actions from psychological studies. Such semantic relations are person-independent; 2) the parameters of the BN model are is learnt from facial expression databases consisting of many different people. Still for many animation applications, the personality of facial expressions is not important since the facial model itself is synthetic. Our goal is to explore the synthesis of the temporal course of facial expressions and faithful facial expression reconstruction. The approach described herein is general enough to achieve this goal.

REFERENCES

[1] Moving Picture Experts Group, “ISO/IEC 14496-MPEG-4 International Standard,” 1998.

[2] P. Ekman and W. V. Friesen, *Facial Action Coding System (FACS): Manual*. Palo Alto, CA: Consulting Psychologists Press, 1978.

[3] P. Ekman and W. V. Friesen, *Unmasking the Face*. New Jersey: Prentice Hall, 1975.

[4] J. Allman, J. T. Cacioppo, R. J. Davidson, P. Ekman, W. V. Friesen, C. E. Lzard, and M. Phillips, “NSF report - facial expression understanding,” tech. rep., Human Interaction Lab, Univ. of California, San Francisco, 1992.

[5] H. Kobayashi and F. Hara, “Facial interaction between animated 3D face robot and human beings,” in *Int’l Conf. Syst., Man, Cybern.*, pp. 3732–3737, 1997.

[6] M. Yoneyama, Y. Iwano, A. Ohtake, and K. Shirai, “Facial expressions recognition using discrete hopfield netural networks,” in *Proc. Int’l Conf. Information Processing*, pp. 117–120, 1997.

[7] C. Padgett and G. Cottrell, “Representing face images for emotion classification,” in *Advances in Neural Information Processing Systems* (M. Mozer, M. Jordan, and T. Petsche, eds.), vol. 9, 1997.

[8] Z. Zhang, M. Lyons, M. Schuster, and S. Akamastsu, “Comparison between geometry-based and gabor wavelets-based facial expression recognition using multi-layer perception,” in *Proc. Int’l Conf. Automatic Face and Gesture Recognition*, pp. 454–459, 1998.

[9] M. J. Lyons, J. Budynek, and S. Akamatsu, “Automatic classification of single facial images,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 21, no. 12, pp. 1357–1362, 1999.

[10] M. Pantic and L. Rothkrantz, "Expert system for automatic analysis of facial expression," *J. Image and Vision Computing*, vol. 18, no. 11, pp. 881–905, 2000.

[11] Y. Yacoob and L. S. Davis, "Recognizing human facial expressions from long image sequences using optical flow," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 6, pp. 636–642, 1996.

[12] M. J. Black and Y. Yacoob, "Recognizing facial expression in image sequences using local parameterized models of image motion," *Int'l J. Computer Vision*, vol. 25, no. 1, pp. 23–48, 1997.

[13] I. A. Essa and A. P. Pentland, "Coding, analysis, interpretation, and recognition of facial expressions," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, no. 7, pp. 757–763, 1997.

[14] N. Oliver, A. Pentland, and F. B*é*rard, "LAFTER: Lips and face real time tracker with facial expression recognition," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, 1997.

[15] Y. Tian, T. Kanade, and J. F. Cohn, "Recognizing action units for facial expression analysis," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, no. 2, pp. 97–115, 2001.

[16] Y. Zhang and Q. Ji, "Active and dynamic information fusion for facial expression understanding from image sequence," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 27, no. 5, pp. 699–714, 2005.

[17] M. Malciu and F. Prêteux, "Tracking facial features in video sequences using a deformable model-based approach," in *Proc. SPIE Int. Soc. Opt. Eng.*, vol. 4121, pp. 51–62, 2000.

[18] J. Chou, Y. Chang, and Y. Chen, "Facial feature point tracking and expression analysis for virtual conferencing systems," in *IEEE Intl' Conf. on Multimedia and Expo*, pp. 24–27, 2001.

[19] T. Cootes, G. Edwards, and C. Taylor, "Active appearance models," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, no. 6, pp. 681–685, 2001.

[20] J. Ahlberg, "An active model for facial feature tracking," *EURASIP Journal on Applied Signal Processing*, vol. 2002, no. 6, pp. 566–571, 2002.

[21] M. Pardas and A. Bonafonte, "Facial animation parameters extraction and expression recognition using hidden markov models," *Signal Processing: Image Communication*, vol. 17, pp. 675–688, 2002.

[22] R. Cowie, E. Dougls-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, and J. G. Taylor, "Emotion recognition in human-computer interaction," *IEEE Signal Processing Magazine*, vol. 2001, no. 1, pp. 33–80, 2001.

[23] A. Raouzaiou, N. Tsapatsoulis, K. Karpouzis, and S. Kollias, "Parameterized facial expression synthesis based on mpeg-4," *EURASIP Journal on Applied Signal Processing*, vol. 2002, no. 10, pp. 1021–1038, 2002.

[24] P. Eister and B. Girod, "Analyzing facial expressions for virtual conferencing," *IEEE Computer Graphics and Applications*, vol. Sep.-Oct., pp. 70–79, 1998.

[25] S. Valente and J.-L. Dougelay, "Face tracking and realistic animations for telecommunicant clones," *IEEE MultiMedia*, vol. Jan.-Mar., pp. 34–43, 2000.

[26] N. P. Chandrasiri, T. Naemura, M. Ishizuka, and H. Harashima, "Internet communication using real-time facial expression analysis and synthesis," *IEEE MultiMedia*, vol. Jul.-Sep., pp. 20–29, 2004.

[27] H. Tao and T. S. Huang, "Facial animation and video tracking," in *Workshop Modeling and Motion Capture Techniques for Virtual Environments*, pp. 242–253, 1998.

[28] F. Lavagetto and R. Pockaj, "The facial animation engine: Toward a high-level interface for the design of MPEG-4 compliant animation faces facial," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 9, no. 2, pp. 277–289, 1999.

[29] T. Goto, M. Escher, C. Zanardi, and N. M-Thalmann, "MPEG-4 based animation with face feature tracking," in *Proc. Erographics Workshop Computer Animation and Simulation*, pp. 89–98, 1999.

[30] S. Kshirsagar, T. Molet, and N. M-Thalmann, "Principal components of expressive speech animation," in *Proc. Computer Graphics Intl*, pp. 38–44, 2001.

[31] Y. Zhang, E. C. Prakash, and E. Sung, "A new physical model with multilayer archtecture for facial expression animation using dynamic adaptive mesh," *IEEE Trans. on Visualization and Graphics*, vol. 10, no. 3, pp. 339–352, 2004.

[32] D. Terzopoulos and K. Waters, "Analysis and synthesis of facial image sequence using physical and anatomical models," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 6, pp. 569–579, 1993.

[33] K. Waters, "A muscle model for animating three-dimentional facial expression," in *Proc. SIGGRAPH*, 1987.

[34] F. I. T. (FIT), "ISO/IEC 14496-MPEG-4 International Standard," 1998.

[35] H. Tao, H. H. Chen, W. Wu, and T. S. Huang, "Compression of mpeg-4 facial animation parameters for transmision of talking heads," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 9, no. 2, pp. 264–276, 1999.

[36] J. Ahlberg and H. Li, "Representation and compressing facial animation parameters using facial action basis functions," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 9, no. 3, pp. 405–410, 1999.

[37] P. Wang and Q. Ji, "Learning discriminant feature for multi-view face and eye detection," in *IEEE Intl' Conf. on Computer Vision*, 2005.

[38] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *European Conf. on Computational Learning Theory*, pp. 23–37, 1995.

[39] H. Gu, Q. Ji, and Z. Zhu, "Active facial tracking for fatigue detection," in *IEEE Workshop on Applications of Computer Vision*, (Florida, USA), 2002.

[40] Z. Zhu and Q. Ji, "Robust pose invariant facial feature detection and tracking in real-time," in *Intl' Conf. on Pattern Recognition*, 2006.

[41] Z. Zhu and Q. Ji, "3D face pose tracking from an uncalibrated monocular camera," in *Intl Conf. on Pattern Recognition*, 2004.

[42] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. San Mateo, CA: Morgan Kaufmann Publishers, 1988.

[43] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proc. of Int'l Conf. on Automatic Face and Gesture Recognition*, pp. 46–53, 2000.

[44] K. P. Murphy, *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.