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

Yongmian Zhang, *Member, IEEE*, Qiang Ji, *Senior Member, IEEE*, Zhiwei Zhu, and Beifang Yi

***Abstract*—This paper describes a probabilistic framework for faithful reproduction of dynamic facial expressions on a synthetic face model with MPEG-4 facial animation parameters (FAPs) while achieving very low bitrate in data transmission. The framework consists of a coupled Bayesian network (BN) to unify the facial expression analysis and synthesis into one coherent structure. At the analysis end, we cast the FAPs and facial action coding system (FACS) into a dynamic Bayesian network (DBN) to account for uncertainties in FAP extraction and to model the dynamic evolution of facial expressions. At the synthesizer, a static BN reconstructs the FAPs and their intensity. The two BNs are connected statically through a data stream link. Using the coupled BN to analyze and synthesize the dynamic facial expressions is the major novelty of this work. The novelty brings about several benefits. First, very low bitrate (9 bytes per frame) in data transmission can be achieved. Second, a facial expression is inferred through both spatial and temporal inference so that the perceptual quality of animation is less affected by the misdetected FAPs. Third, more realistic looking facial expressions can be reproduced by modelling the dynamics of human expressions.**

***Index Terms*—Bayesian networks (BNs), facial animation, facial expression synthesis, MPEG-4 facial animation parameters (FAPs).**

I. Introduction

FACIAL expression synthesis is of interest for many multimedia applications such as human–computer interaction (HCI), entertainment, virtual agents, video teleconferences, and avatars. Current technologies are still unable to synthesize human expressions in a realistic and efficient manner and with crucial emotional contents. Since the MPEG-4 visual 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 a computational study of facial expressions. The MPEG-4 visual standard provides an alternative way of modelling facial expression and the underlying emotion, which are strongly influenced by psychological studies such as Ekman's facial action coding system (FACS) [2]. The FACS has now become the de facto standard in characterizing facial expressions.

The MPEG-4 visual standard specifies a set of facial definition parameters (FDPs) and facial animation parameters (FAPs) for facial animation. The 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 ethnics-independent facial action units (AUs) [3]. The FAPs are adequate to define the measurement of muscular actions relevant to AUs. Moreover, the FAPs can be placed on any synthetic facial model in a consistent manner with little influence by the inter-personal variations. The FDPs are normally transmitted once per session and then followed by a stream of compressed FAPs. The animation of a virtual face is achieved by first transmitting the coded FAPs and then resynthesizing on the client-side. To accommodate very low bandwidth constraint, the FAPs must be compressed so that they can be transmitted in very low bitrate.

Despite significant progress, the current techniques in facial expression synthesis face several issues that still need to be resolved.

1) Although the discrete cosine transform (DCT) technique can achieve a high FAP compression, the DCT involves a large coding delay (temporal latency) that makes it unsuitable for real time interactive applications. The principal component analysis (PCA) is able to achieve a high FAP compression for intraframe coding, however, it compromises the reconstruction accuracy.
2) An automated video analyzer may often misdetect some facial features. Consequently, this may create animation artifacts on the facial model, which affects the perceptual quality of facial animation.
3) The intensity of facial expressions reveals the emotional evolution. It is difficult for machine to extract the subtle variation of facial features. Consequently, a dynamic behavior of human expressions is difficult to be animated. However, as indicated by physiologists, the temporal course information is necessary for life-like facial animation [4].
4) The AUs are the linguistic descriptions of facial muscle activities from psychological view, while the FAPs provide a way in defining the measurement of muscular actions. However, there is a lack of computational model to integrate the AUs and the FAPs systematically.

This work is to introduce an alternative approach to address the above issues. The proposed approach allows faithful visual reproduction of dynamic human expressions on a synthetic face

Manuscript received December 20, 2005; revised May 2, 2006 and May 5, 2007. First published xxx; current version published xxx. This work was supported in part by the Air Force Office of Scientific Research under Grant F49620-03-0160. This paper was recommended by Associate Editor G. Wen.

Y. Zhang and Q. Ji are with the Department of Electrical, Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180 USA (e-mail: zhangy@ecse.rpi.edu; qij@ecse.rpi.edu).

Z. Zhu is with **[AUTHOR: PLEASE PROVIDE A COMPLETE MAILING ADDRESS.—ED.]** Sarnoff Corporation, Princeton, NJ USA.

B. Yi is with the Department of Computer and Information Sciences, State University of New York, Fredonia, NY 14063 USA (e-mail: Beifang.Yi@fredonia.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2008.928887

model using the MPEG-4 FAPs, particularly, for low bitrate interactive applications such as videophone systems and a mobile terminal using a cellular network. Our work has the following three main contributions, which sets it apart from our previous work [5] and the work from other researchers presented in Section II. The first contribution is a computational model that systematically integrates the FAPs and the AUs into a probabilistic framework. The second contribution is a coupled Bayesian network that allows to unify the facial expression analysis and synthesis into one coherent structure to perform consistent reasoning, feature fusion, and FAP reconstruction. Therefore, the temporal course of a facial expression can be animated to achieve a lifelike animation. In addition, by taking advantage of Bayesian inference in handling missing data, the robust reconstruction of the facial expression is possible even in the absence of some FAPs. The third contribution is to achieve a very low bitrate for visual reproduction of facial expressions at the synthesizer. With a coupled BN, data communication between the analysis end and the synthesizer can be implemented as the dependency between the two BNs. Therefore, facial expressions are reproduced at the synthesizer without recourse to streaming the FAPs. Instead, we transmit only 9 bytes of data per frame to the synthesizer (6 bytes for the state of the six facial expressions and 3 bytes for face pose).

Using a coupled Bayesian network to analyze and synthesize the dynamic behavior of facial expressions is the major novelty of this work. The remainder of this paper is organized as follows. Section II reviews related works. Section III provides a brief overview of our system. Section IV gives video analysis. Sections V and VI cover our approach in facial expression analysis and synthesis. We present experiments in Section VII. The final section provides discussions and conclusions.

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.

Automatic facial expression recognition had an early start with static face images [6]–[9]. There have been several attempts to recognize facial expressions over time from video sequences. Yacoob and Davis [10] proposed a region tracking algorithm to integrate spatial and temporal information at each frame in an image sequence. Black and Yacoob [11] used local parameterized flow models to identify facial expressions. An affine model and a planar model represent head motion, and a curvature model represents non-rigid facial motion. In [12], Essa and Pentland used the invariance between the motion energy template learned from ideal 2-D motion views and the motion energy of the observed image to classify the facial expressions. Oliver *et al.* [13] applied hidden Markov model (HMM) to recognize mouth-related expressions. The facial expression is identified by computing the maximum likelihood of the input sequence with respect to all HMMs which are trained for each expression. Tian and Kanade [14] used the two separate neural networks to recognize the upper face AUs and the lower face AUs. Recently, Zhang and Ji [5] proposed to use dynamic Bayesian networks for modelling both spatial and dynamic relationships among facial expressions, and for recognizing the six basic facial expressions through a probabilistic inference. The MPEG-4 visual standard has motivated intensive research in facial feature extraction for facial animation [15]–[18]. Substantial efforts in facial expression analysis with MPEG-4 FAPs have been made recently [19]–[21]. Among these works, either rule-based techniques or HMMs are used. The rule-based approach lacks the expressive power to capture the temporal behaviors and dependencies among facial actions. The HMMs are able to model time series with uncertainty, but they cannot represent variables at different levels of abstraction and the dependencies among facial actions. To overcome these limitations, in this paper we incorporate the Bayesian facial expression model from our previous work [5] with MPEG-4 FAPs.

In the area of multimedia, researchers have shown great interest in lifelike animated agents with realistic behavior. Eisert and Girod [22] applied facial expression synthesis to virtual conference, whereby the optical flow is used to capture the motion information to estimate 17 FAPs for controlling a virtual head. A similar approach is presented by Valente and Dougelay [23]. In facial animation, Tao and Huang [24], Lavagetto and Pockaj [25], Goto *et al.* [26], Raouzaiou *et al.* [20], and Kshirsagar *et al.* [27] 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. A thorough overview of MPEG-4 visual standard as related to facial animation technology can be found in [28]. The facial AUs of the FACS are often used to group the muscle activities in facial animation. For examples, Zhang *et al.* [29] and Terzopoulos and Waters [30], [31] group muscles into AUs by their positions in their facial animation system.

The FAP compression can be categorized as intraframe coding and interframe coding. In intraframe coding, one way to achieve data reduction is to send only a subset of active FAPs to a synthesizer. The MPEG-4 visual standard [1] 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.* [32] and Ahlberg and Li [33] use the 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. Two interframe coding schemes, predictive coding (PC) and discrete cosine transform (DCT), are adopted in the MPEG-4 animation technology. 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 in smaller quantities, fewer bits are needed to represent these differences. If the FAP sampling rate is high (> 10 Hz), the DCT technique may be used. By performing the DCT in each temporal segment, a high compression efficiency can be achieved; but it introduces a coding delay.

III. System Overview

Our methods have been integrated into an unified system for facial expression analysis and synthesis. Fig. 1 gives a block diagram to show the system components. Our system is suitable for interactive applications such as teleconferencing and videophone that require a live facial expression not only to be transmitted in very low bitrate, but also to be reproduced realistically

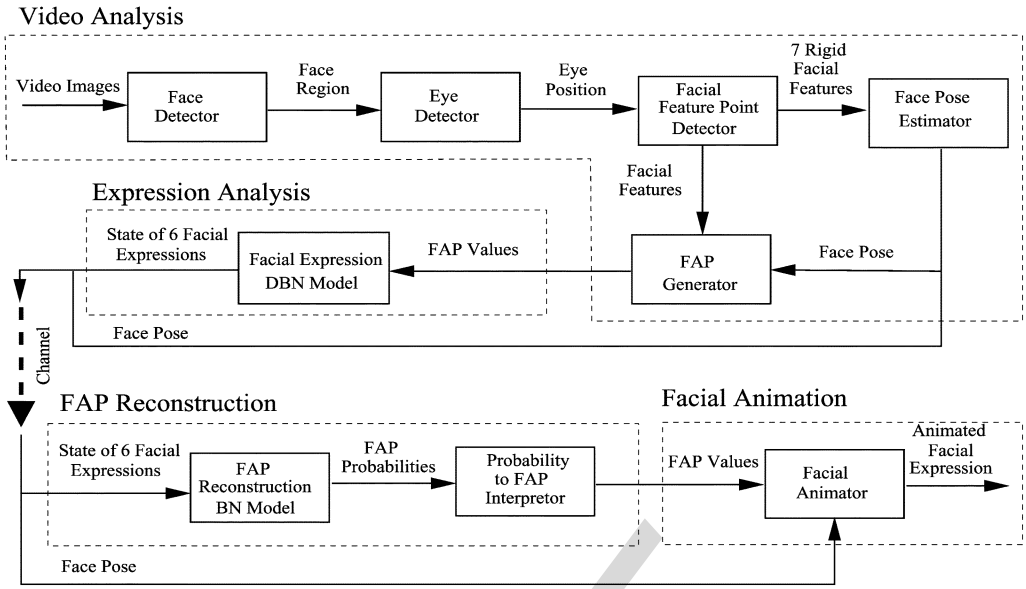

Fig. 1. Block diagram of our facial expression animation system, including the major modules and their relationships. The channel here represents a very low bitrate communication link. The state of six facial expressions is a probability distribution of the six facial expressions.

and faithfully at a remote receiver. We 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 the measurements of FAPs and face pose. The use of 3-D facial 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 detected facial feature points are used to produce measurements for face pose and the FAPs as defined in MPEG-4 visual standard.

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

FAP Reconstruction: Using the probability distribution of six facial expressions produced by the analysis end, the synthesizer reconstructs the FAPs and their intensity through a static Bayesian network (BN) to provide quantitative information about the facial expressions and their temporal evoluation.

Facial Animation: This module uses the reconstructed FAPs to reproduce the facial expression on the facial model. The dynamics of facial expressions is characterized by the intensity development of the reconstructed FAPs.

Our system has three major advantages: 1) very low bitrate 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 incorrectly or undetected FAPs at the analysis end; and 3) a realistic and visually faithful facial expression can be reproduced by modelling the dynamic behavior of human expressions.

The software is developed for FAP extraction and face pose tracking. This software is real time, fully automatic, and applicable to different people. The 3-D synthetic head model consisting of 3000 vertexes and 3200 polygons is developed. We utilize Intel's Probabilistic Networks Library 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 3-D 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 to simulate a data communication channel, exactly as would be performed in an actual application.

IV. Video Analysis

Here, we first give a brief introduction to MPEG-4 visual standard related to facial expression animation (readers may refer to [28] for the details of the MPEG-4 visual standard). We then describe our approach in FAP extraction and face pose estimation.

A. Facial Animation Parameters

The FAPs are a set of parameters defined in the MPEG-4 visual standard [1] for the animation of synthetic face models. There are 68 FAPs including 2 high-level FAPs used for visual phoneme and expression, and 66 low-level FAPs used to characterize the facial feature movements over jaw, lips, eyes, mouth, nose, cheek, ears, etc. We select 27 FAPs which are only active in facial expressions to characterize the six basic facial expressions as summarized in Table I. The FAPs are computed through tracking a set of facial features defined in Fig. 2, and they 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

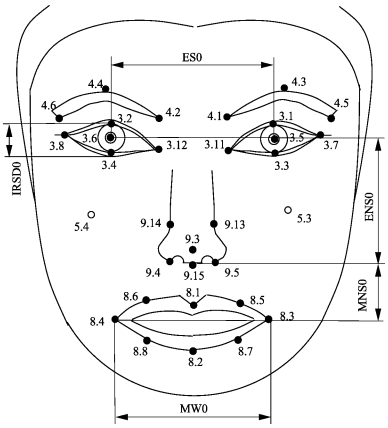

Fig. 2. Neutral face model and feature points used to define FAPUs. The feature points are numerated with MPEG-4 visual standard. Only the feature points marked with solid dots are tracked.

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 UNITS

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

(MNS0), and lip corners (MW0), as shown in Fig. 2. The facial feature points in Fig. 2 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. 2) are not trackable due to their unreliability in tracking, but they can be inferred in FAP reconstruction, based on their relationships with other FAPs.

B. Facial Feature Detection

To extract the FAPs, facial feature points have to be detected as 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

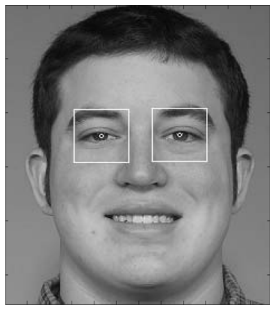

Fig. 3. Example of detected eyes and pupils (marked with white circles) by using the Adaboost classifier and Haar features.

in [34]. Both face and eye detector use an AdaBoost algorithm [35] with non-linear discriminate features. Fig. 3 shows an example of the 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

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

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 $k_i \in \{(\pi/2),(\pi/4),(\pi/8)\}$, and six orientations (ϕ) from 0 to π differing by $(\pi/6)$. 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. The details about this work may be found in [36] and [37].

C. Face Pose Estimation

The objective of face pose estimation has twofold: 1) the face pose may distort the FAPs if they are computed directly from the 2-D images, and such distortion needs to be eliminated in the facial expression analysis and 2) the face pose itself 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 3-D face shape model and seven relatively rigid (or near rigid) points including four eye corners and three points on the nose as control points to determine the 3-D head movement, as shown in Fig. 4. Given the detected image coordinates for the seven points and their corresponding coordinates in the 3-D model, the 3-D face pose, i.e., the pan, tilt and swing angles (ω, ϕ, κ) as well as a scale factor can be estimated using a robust pose estimation technique, assuming weak perspective projection. Details about the pose estimation technique may be found in [37]. The allowed out-of-plane head rotation is around $\pm 40^\circ$. Fig. 5 illustrates a facial tracking example, where the face normal perpendicular to

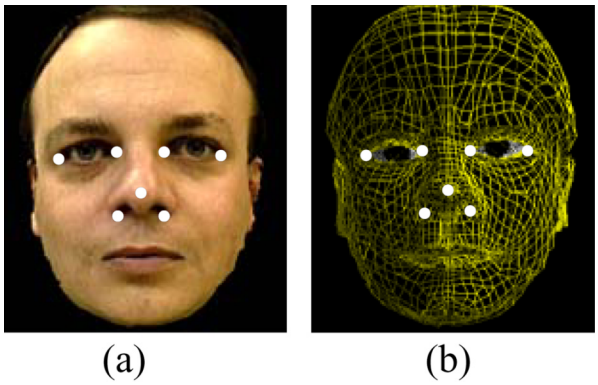

Fig. 4. 3-D facial shape model. (a) Frontal face image. (b) 3-D face shape model with seven rigid facial features (marked with white dot) as control points.

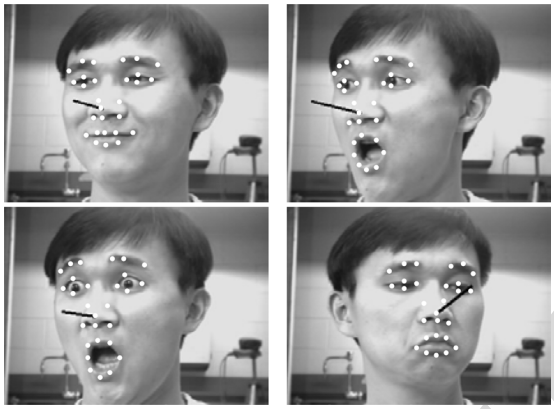

Fig. 5. Illustration of facial feature and pose tracking under facial expressions. Here, the face normal is represented by a dark line and the detected features are marked with white dots.

the face plane is computed from the three estimated Euler face pose angles.

D. FAP Generation

Once the face pose is estimated, the 3-D coordinate of any facial feature can be recovered by eliminating the face pose effect from its 2-D coordinate in the image. Specifically, let (u_i, v_i) and (x_i, y_i, z_i) be the coordinates of the facial feature point i in 2-D and 3-D respectively. Since the z_i coordinate value of each 3-D facial feature is adapted from a neutral generic 3-D face model directly, (x_i, y_i) can be recovered as follows:

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

where $(u_0, v_0)^T$ and $(x_0, y_0, z_0)^T$ are the centroids of the seven points in 2-D and 3-D, respectively, and M is the pose matrix determined by the 3 pose angles. M can be parameterized as

$$M_{2\times 2} = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix}. \tag{3}$$

Using (2), the 3-D 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 3-D coordinate of each facial feature, the associated FAPs can be computed directly. A FAP can be quantified by the spatial distance between the corresponding facial feature points as given in Table III, which is measured by either D_x or D_y, where D_x or D_y is the distance of two 3-D 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 $D_y = |y_1 - y_0|$.

TABLE III
RELATIONSHIPS 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. FAP5_, 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. 2. FAP41,42 are not tracked in our feature detector. 3. $D_x(p_1, p_2)$ and $D_y(p_1, p_2)$ are the distance of two points p_1 and p_2 (as defined in Fig. 2) in X and Y direction respectively.

V. FACIAL EXPRESSION ANALYSIS

Here, we first give a descriptive model of facial expressions by using the AUs and the 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 [5]. 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, while an auxiliary AU is the one that can only be

TABLE IV
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.

combined with the primary AUs as supplementary cue in distinguishing facial expressions. For example, AU9 (Nose Wrinkler) can be directly categorized as disgust, but it is ambiguous to categorize AU17 (Chin Raiser) as disgust. When AU9 and AU17 combine, this AU combination is more certain to be the expression of disgust. Thus, AU9 is a primary AU of disgust while AU17 is an auxiliary AU. Table V gives a summary of the primary AUs and the auxiliary AUs associated with the six facial expressions. Such a categorization of AUs represents an extension to Ekman's work [2].

To automatically quantify the activation of the muscles directly from a face image, we need quantitatively relate AUs to facial feature movements. The FAPs provide a way in measuring facial feature movements. By relating FAPs to AUs, FAPs can be used to quantitatively characterize the muscle movement specified by an AU. Table III gives the relations between the FAPs and AUs.

B. A Computational Model of Facial Expressions

Tables III and V deterministically characterize the relations between facial expressions and the AUs and between the AUs and the FAPs. To account for the uncertainty in the feature measurement and the dependency among the AUs, we cast the deterministic relations into a probabilistic framework using a Bayesian network (BN) [38], which provides us a mathematically rigorous foundation for consistent, coherent and efficient reasoning and visual information fusion.

Our BN model of facial expressions has three different abstractions: expression layer, facial AU layer and FAP layer as shown in Fig. 6. 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, respectively. 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 relations between the AUs and facial expressions as given in Table V. The FAPs occupy the lowest level of layers and they are observable variables in the model.

The values of FAPs are estimated by the positions of the detected facial features, and their values (amplitude) are divided 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 facial expression database [39]. Notice that the intensity of FAPs does not represent the intensity of the intended facial expression. The emotional intensity for an expression is measured by its probability. The conditional probabilities required to parameterize the BN model are trained from 50 subjects in the Cohn-Kanade database.

To capture the temporal evolution of a facial expression, the static BN model is further extended with the DBNs. Our DBN model of facial expressions is made up of interconnected time slices of a static BN, and the dependency between two neighboring time slices are based on a first-order HMM. The DBNs enable to correlate and associate the continual arriving evidences through temporal dependencies to perform reasoning over time. Specifically, let Θ be a hypothesis variable of facial expression, let $S^{(1)}, \ldots, S^{(n)}$ be the n intermediate variables in the DBN, and let $\mathbf{e}$ 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

$$P(\Theta_t|\mathbf{e}) = \frac{P(\Theta_t, \mathbf{e})}{\sum_{\Theta_t} P(\Theta_t, \mathbf{e})} \tag{4}$$

where t is the discrete time index and

$$P(\Theta_t, \mathbf{e}) = \sum_{\Theta_{t-1}} \sum_{S_t^{(i)}} \Bigg\{ P(\Theta_t|\Theta_{t-1}) P(\Theta_{t-1}) \times \prod_i P\left(S_t^{(i)}\right) |\pi\left(S_t^{(i)}\right) P(\mathbf{e}|\pi(\mathbf{e})) \Bigg\}, \quad i = 1, \ldots, n \tag{5}$$

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 (5) 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. Details about our DBN model and its construction may be found in [5].

VI. FACIAL EXPRESSION SYNTHESIS

In the MPEG-4 facial animation technology, the FAPs have to be transmitted to the synthesizer in order to reproduce the facial expressions on the client-side. To accommodate low bandwidth constraint, the FAPs must be compressed. There are several shortcomings with directly transmitting FAPs. First, the misdetected FAPs in the analysis end will create strange animation artifacts, which directly affect the animation perceptual quality.

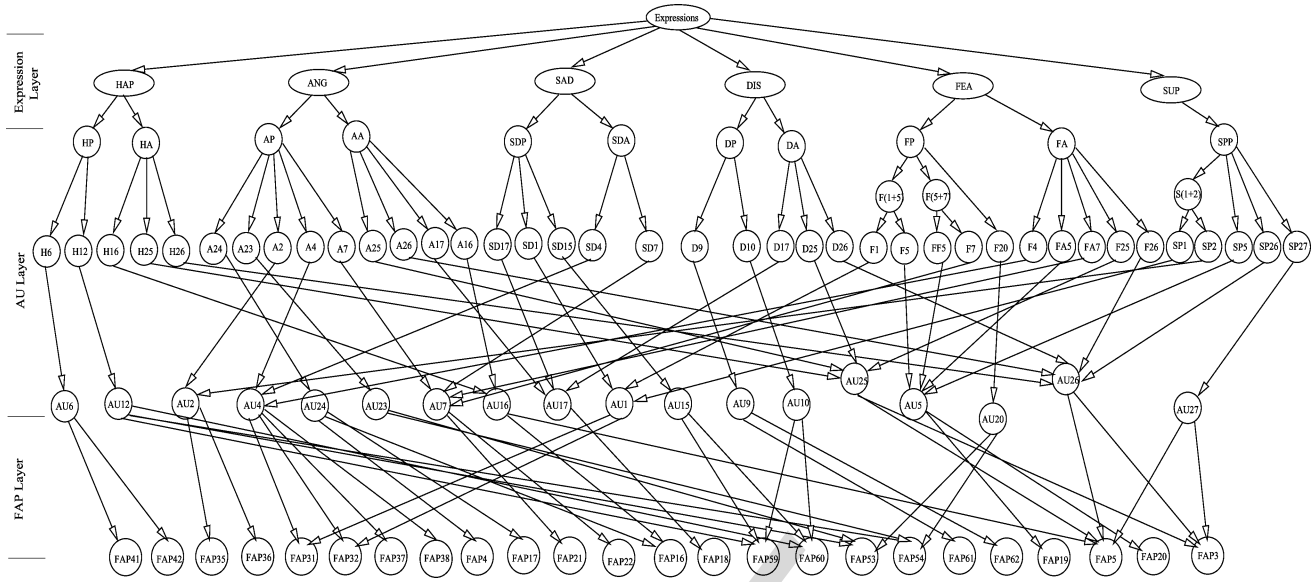

Fig. 6. BN model of the six facial expressions. The notations HAP, ANG, SAD, DIS, FEA, and SUP denote Happiness, Anger, Sadness, Disgust, Fear, and Surprise, respectively. HP, AP, SDP, DP, FP, and SPP denote the primary AUs of happiness, anger, sadness, disgust, fear, and surprise, respectively. HA, 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.

Second, some FAPs such as FAP41 and FAP42, which are unmeasurable by video analyzer due to the difficulty in detecting facial feature on the cheek (see Fig. 2), also affect the animation perceptual quality. Third, for interactive applications with very low bandwidth constraint, the PCA technique may be used for efficient FAP compression, but it compromises FAP reconstruction accuracy. This work aims to overcome these problems.

At the synthesizer, we use a static BN that has the same spatial structure as the DBN model in facial expression analysis (see Fig. 6). The 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. 7 depicts the dependency graph of such a coupled BN. The BN model at the synthesis end is coupled with the DBN at analysis end by the conditional dependency link between their top nodes Θ and Θ' such that the probability distribution of the six facial expressions at analysis end passes to the BN at the synthesizer. Such a link acts like a data stream communication channel in the actual applications. At the analysis end, the hypothesis node Θ (the probability distribution of the six facial expressions) completely summarizes the continual arriving visual evidences (FAPs) by integrating them through Bayesian inference. At the synthesizer, the FAPs are inferred given Θ' through a top-down predictive inference. Therefore, in the actual applications, we only need to transmit the state of Θ (the probability distribution of the six facial expressions) and the values of the pose angles to the synthesizer for reconstructing the FAPs. Since 1 byte provides sufficient accuracy to represent each of the six expressions or each of the three face pose angles, we need to transmit only 9 bytes of data per frame.

BNs can also be used for causal reasoning to specify how causes generate effects. Specifically, through a top-down inference, we can compute the probabilities of a set of FAPs given

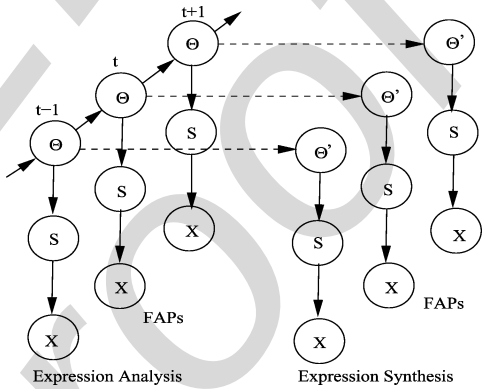

Fig. 7. Dependency graph of a coupled Bayesian network. The facial expression analysis end uses a DBN and the facial expression synthesizer uses a static BN (see Fig. 6 for the detail of a BN model). The nodes Θ, Θ', S, and X denote the hypothesis nodes, a set of intermediate nodes, and a set of FAPs, respectively.

Θ', which equals to Θ if the data is completely transmitted to the synthesizer from the analyzer. Now let X_j be a specific FAP and let $S = \{S_1, \ldots, S_n\}$ be a set of intermediate nodes interconnecting Θ' and X_j. Given a set of visual readings $\mathbf{e}$ (which represents FAP measurement) at the analysis end, the probability of a specific FAP X_j at the synthesis end may be computed by

$$P(X_j|\mathbf{e}) = \sum_{\Theta', S_i} \left\{ P(X_j|\pi(X_j)) \prod_{i=1,\ldots,n} P(S_i|\pi(S_i)) \times P(\Theta'|\mathbf{e}) \right\}, \quad (6)$$

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

$$P(\Theta'|\mathbf{e}) = \sum_{\Theta} P(\Theta'|\Theta)P(\Theta|\mathbf{e}) \tag{7}$$

then we have

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

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 FAPs. 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 $P(X_j = \text{true}|\mathbf{e})$ to reconstruct a FAP and its intensity. Again, the conditional probabilities of the BN model for both the analysis and synthesis end are learned from facial expression databases. We can see from (8) that the FAP intensity, i.e., the probability of a FAP, develops as a function of $P(\Theta|\mathbf{e})$, where $P(\Theta|\mathbf{e})$ results from integrating the continual arriving visual evidences over time. At the synthesis end, the FAPs most relevant to the current state of facial expressions have a higher probability than others. In other words, the FAP intensity at the synthesis end evolves according to 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, the misdetected FAPs or the unmeasurable FAPs such as FAP41 and FAP42 at the analysis end can still be inferred 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 the six facial expressions $P(\Theta|\mathbf{e})$ and three pose angles so that very low bitrate (9 bytes per frame) in data transmission can be achieved.
3) Unlike transmitting the FAPs directly to the synthesizer, our approach is less affected by feature detection errors, e.g., a FAP that may not be extracted for several frames or one of mouth corner is misdetected. The missing FAPs can be inferred based on their semantic relations to other FAPs so that the perceptual quality of facial animation does not suffer.

Now, we need to tranlate a FAP intensity to a FAP amplitude in order to drive the facial animation. Let f be FAP amplitude and $f_{\max}$ be its maximal amplitude. Let p_n, p_c and p_a be the intensity of this FAP when a facial expression is in neutral state, current state and apex state, respectively. Then, the amplitude of a FAP involved in a specific facial expression can be simply computed as

$$f = \frac{p_c - p_n}{p_a - p_n} f_{\max} \tag{9}$$

TABLE VI
HAPPINESS-RELATED FAP INTENSITY AT NEUTRAL AND APEX, AND THEIR MAXIMAL AMPLITUDE

FAPs	Neutral (p_n)	Apex (p_a)	Maximal Amplitude (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)

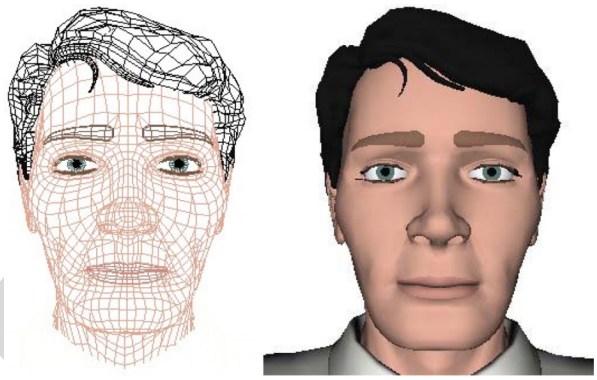

Fig. 8. (a) Wireframe facial model. (b) Synthetic facial model.

where $f_{\max}$ can be predetermined based on the available facial expression database, and f_n and f_a can be obtained from the BN model, as shown in Table VI. Since $f_{\max}$ is measured by FAPU, it allows us to map f on any facial model.

Given the FAP amplitudes, a facial expression can be reproduced on a synthetic facial model. Our 3-D synthetic facial model consists of 3000 vertices and 3200 polygons to represent a generic face, as shown in Fig. 8. The facial model rotates around its center given 3-D 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 points move, i.e., the trajectory of feature points as a function of the FAP amplitude. 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 a linear approximation of vertex motion given FAP amplitudes so that deformable lattices can be generated on the facial model. Fig. 9 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. Finally, the facial model is rotated according to the 3-D pose (ω, ϕ, κ) received from the analysis end.

Our approach to the visual reproduction of facial expressions and face pose is summarized as follows.

Step 1) Obtain visual measurements of FAPs $\mathbf{e} = \{e_1,\ldots,e_n\}$, and face pose (ω, ϕ, κ) by the video analyzer.

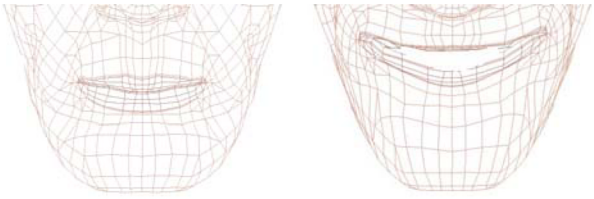

Fig. 9. Wireframe of the mouth in (a) neutral and (b) the apex of happiness and the deformable vertices around the mouth. Given an FAP (here raising mouth corner), the deformation of vertices around the mouth corners is determined by a linear interpolation table.

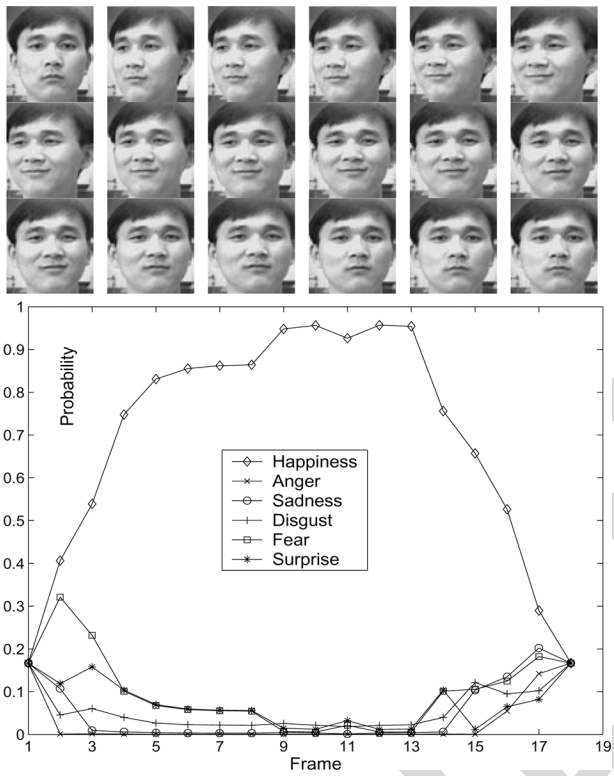

Fig. 10. Top: a short image sequence showing that the subject's expression starts with the neutral, then gradually reaches the apex and finally releases. Bottom: the probability distribution generated by our facial expression analysis model (5). The expression probability development indeed corresponds to the temporal evolution of the facial expressions.

Step 2) compute the probability distribution of the six facial expressions $P(\Theta|\mathbf{e})$ by DBN inference given $\mathbf{e}$.

Step 3) Transmit $P(\Theta|\mathbf{e})$ and (ω, ϕ, κ) to the synthesizer.

Step 4) Compute the probability of FAPs $P(X_i = \text{true}|\mathbf{e})$ by BN top-down inference given $P(\Theta|\mathbf{e})$, $i = 1 \ldots n$.

Step 5) Obtain FAP amplitude f_i based on $P(X_i = \text{true}|\mathbf{e})$, $i = 1 \ldots n$.

Step 6) Animate facial expression and orientation by mapping feature point motion onto vertex motion given a group of FAP amplitudes $\mathbf{f} = \{f_1, \ldots, f_n\}$ and face pose (ω, ϕ, κ). Go to step 1).

VII. Experiments

Here, we first show how the dynamic nature of facial expression is modelled and then we perform the tests of facial expression synthesis and animation. Finally, a performance evaluation of the FAP reconstruction quality is given.

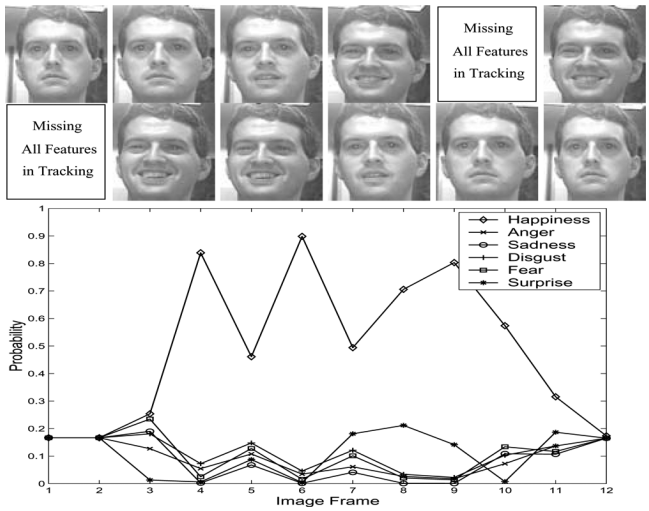

Fig. 11. Top: An image sequence assuming that the facial features in some image frames are missed; Bottom: the intensity scale of the six facial expressions from our facial expression analysis model. The valleys of the curve at frames 5 and 7 are caused by the absence of certain facial features.

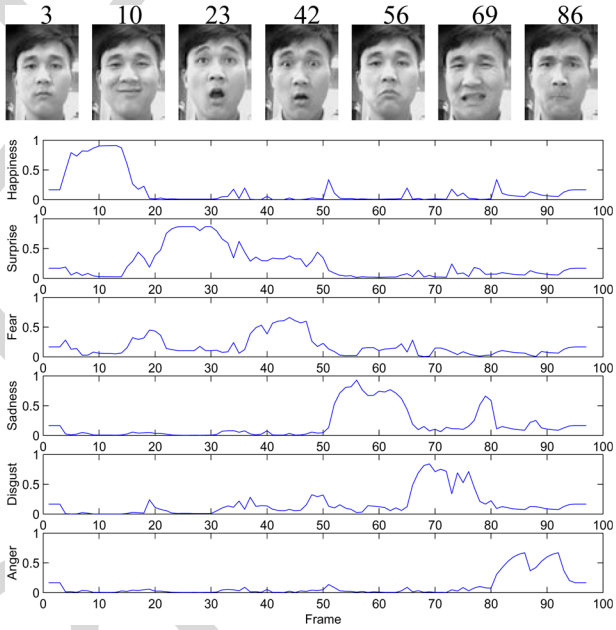

Fig. 12. Top: a video sequence containing the six facial expressions and only eight snapshots are shown for illustration. Bottom: the emotional intensity (probability distribution over the six facial expressions) plotted over time.

A. Facial Expression Analysis

With regard to the intensity of the facial expression, the muscle contraction rates need to be measured at every stage of the emotional development. However, since there are inter-personal variations with respect to the amplitudes of facial actions, it is practically difficult to determine the expression intensity of a given subject by machine estimation. The duration of an emotional expression is often related to the emotional intensity. This means that the current state of a facial expression can be inferred relying on the combined information of current visual

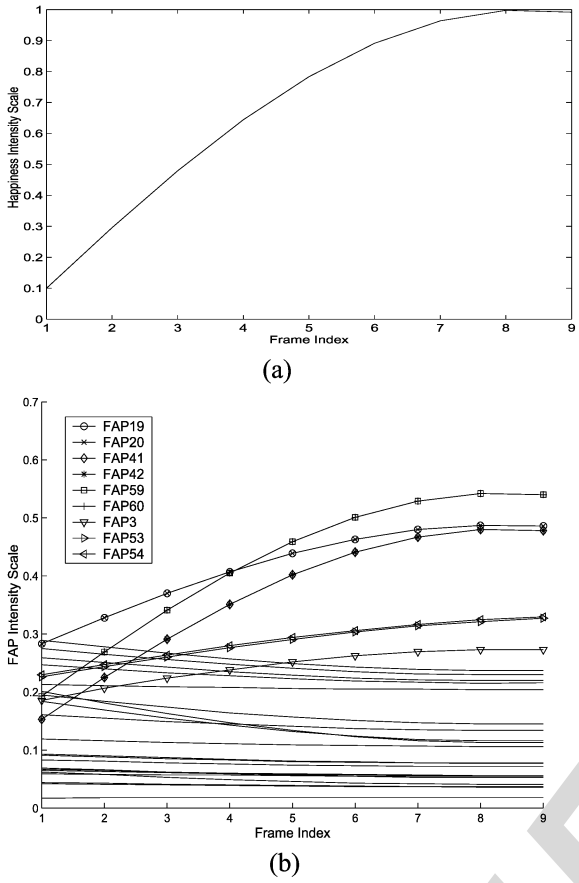

Fig. 13. (a) Perfect intensity curve of happiness from the neutral state to the apex. (b) Intensity of the FAPs evolves in accordance with the intensity of the facial expression. Only the FAPs associated with this expression are marked with line symbols. Notice that, because of the symmetric nature of human face, the curves of FAP19, 41, 59, and 53 are overlapped with the curves of FAP20, 42, 60, and 54, respectively.

cues as well as the preceding evidences. Hence, as we can observe from a typical result in Fig. 10, the evolution of emotional magnitude can be well modeled; this enables us to represent the dynamics of the six basic emotional facial expressions.

Through its temporal inference, the DBN facial expression model can effectively infer facial expression at current frame not only based on current feature measurements but also based on previous feature measurements. Hence, facial expression can still be reasonably recognized despite of the misdetection of certain facial features at current frame. In addition, the presence of other facial features and their built-in relationships to the missing features in BN further help estimate the facial expression. This can be seen from Fig. 11.

Fig. 12 illustrates an output showing a temporal course of facial expressions. The images are sampled in every seven frames from a 700-frame sequence (captured at 30 f/s) containing the six facial expressions and the neutral state. Although, as we can see from the figure, there are a certain number of recognition errors due to feature detection errors, a visual inspection indicates that the expression evolvement is well reconstructed. A quantitative performance evaluation of recovering the dynamics of facial expressions with DBN can be found in our previous work [5]. The ability of our approach to correlate and reason about

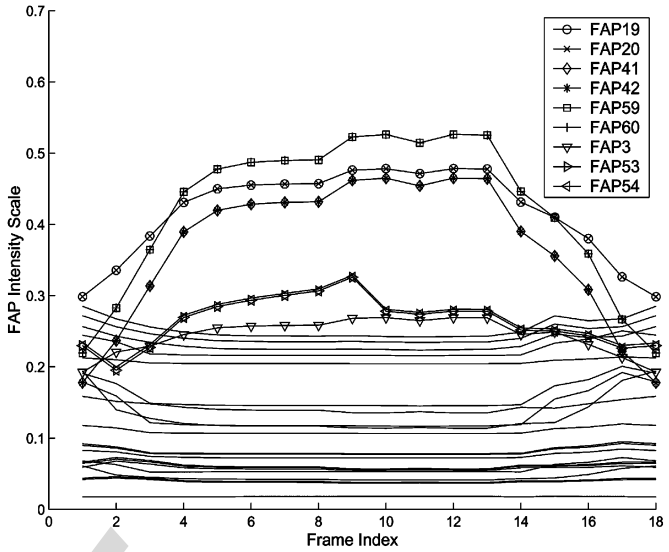

Fig. 14. Intensity of reconstructed FAPs evolves in accordance with the intensity of a facial expression. Only the FAPs associated with the facial expression are marked with line symbols. Notice that, because of the symmetric nature of human face, the curves of FAP19, 41, 59, 53 are overlapped by the curves of FAP20, 42, 60, and 54, respectively. The expression analysis result is from Fig. 10.

facial temporal information over time allows to capture the dynamic behavior of facial expressions in an image sequence such that various stages of the emotional development can be analyzed by machine. This enables us to reproduce a realistic behavior of facial expressions at the synthesis end.

B. *Facial Expression Animation*

First, we study the FAP reconstruction quality. We assume that we have an ideal probability distribution of facial expressions received from the analysis end, as shown in Fig. 13(a), which reflects a facial expression starting from its neutral state to its apex. Notice that for clarity we only shows the intensity of happiness in this figure. Fig. 13(b) illustrates the intensity of reconstructed FAPs, which shows that the FAP intensity evolves as the intensity of the happy expression increases. We can see from the figure that the reconstructed FAPs relevant to the happy expression (FAP3, 19, 20, 41, 42, 53, 54, 59, and 60) dominate other FAPs. This agrees with Tables III and V. In addition, though the feature points 5.3 and 5.4 on the face cheek (see Fig. 2) are not trackable, FAP41 and FAP42 represented by these features can still be inferred to certain degree through their semantic relationships to other tracked facial features.

Now, we use the result from facial expression analysis as shown in Fig. 10 to illustrate the facial animation. Fig. 14 depicts the reconstructed FAP intensities given the facial expression in Fig. 10. A visual inspection shows that the reconstructed FAP intensities agree with the evolution of this facial expression. Based on the reconstructed FAPs in Fig. 14, we reproduce the temporal course of this expression and the face pose on a facial model as shown in Fig. 15. An additional example is given in Fig. 16. Visual inspection of a number of image sequences for different subjects reveal that the proposed method can well synthesize the temporal development of a facial expression. To

Fig. 15. Example result of synthesized temporal course of a facial expression for a given facial expression from facial expression analysis in Fig. 10.

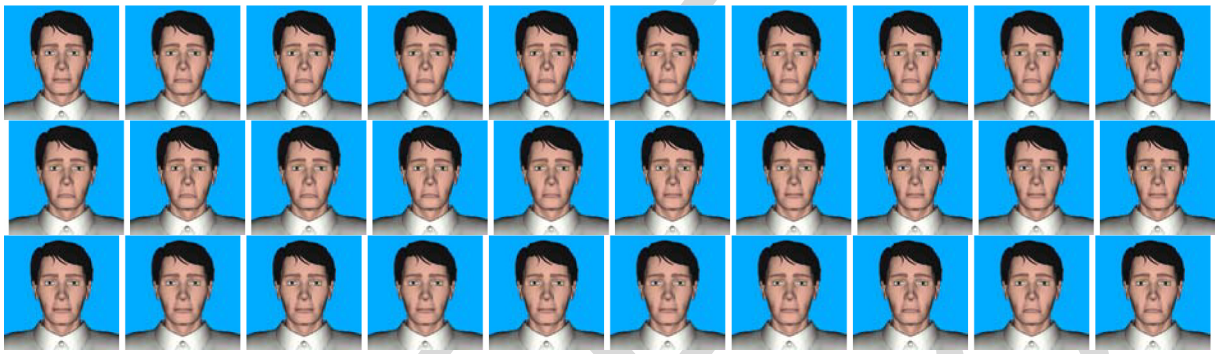

Fig. 16. Animation result for the temporal course of sadness. The facial expression analysis results from Fig. 12 (frames 50 to 70).

better characterize the performance of our method, a more quantitative and systematic performance evaluation is needed. Such a study, however, requires a quantitative metric to characterize the quality of both the reconstructed facial expression as well as its temporal development. We will pursue this study in the future.

The proposed approach is unable to animate the individuality of a facial expression. For example, the subject smiles with jaw closed in Fig. 10, while the jaw is slightly open in the synthesized result as can be seen in Fig. 15. In the facial expression model, the cause and effect relations among the FAPs and facial expressions are determined uniquely, based on Ekman's linguistic description of facial expressions [2]. According to this description, when a subject performs smile, the mouth slightly opens and the extent of mouth open depends on the expression intensity. Therefore, a smile without opening the jaw is like FAP3 (jaw open) is not generated in our case. As mentioned previously, FAP3 can still be inferred reasonably by its semantic relationships to other features as shown in Fig. 15. This is a useful feature of our method, i.e., a missing FAP at the analysis end can still be recovered to certain degree at the synthesis end, which helps improve the animation perceptual quality. One solution to personalize the animation result is to build a facial expression model for each individual; but it is less useful in practice. As indicated by physiologists in [4], time course information is necessary for 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. This paper aims to achieve a more realistic facial animation by explicitly modeling the temporal development of facial expressions.

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 quality of FAP reconstruction. However, our approach is to model the temporal development of facial expressions to achieve more life-like facial animation. To accomplish this, we use BNs to fuse the FAPs over time at the analysis end and to reconstruct them at the synthesizer. As this approach does not stream directly the FAPs to the synthesizer, the original and the reconstructed FAPs are expected to differ. Thus, the PSNR cannot be used for the performance evaluation in this case. Instead, we propose to evaluate the fidelity of facial expression reconstruction with the following aspects: 1) the dynamic effect of resulting facial animation, 2) the quality of facial expression reconstruction, and 3) the robustness under FAP measurement errors.

To observe how the FAPs evolve, we manually create a perfect distribution of facial expressions, as shown in Fig. 17. The

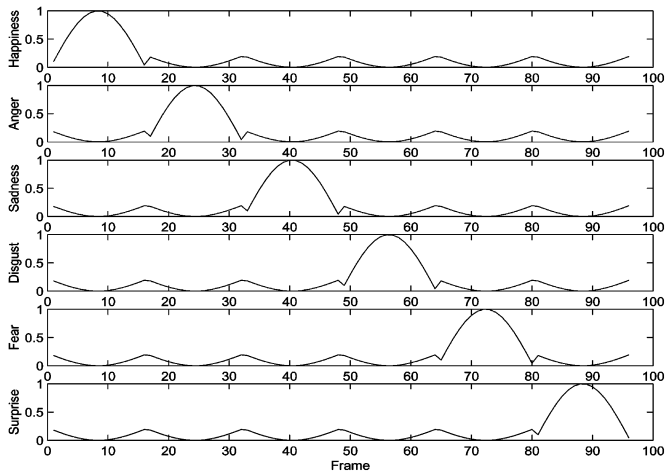

Fig. 17. Manually created intensity curve depicts an ideal temporal course of the six facial expressions.

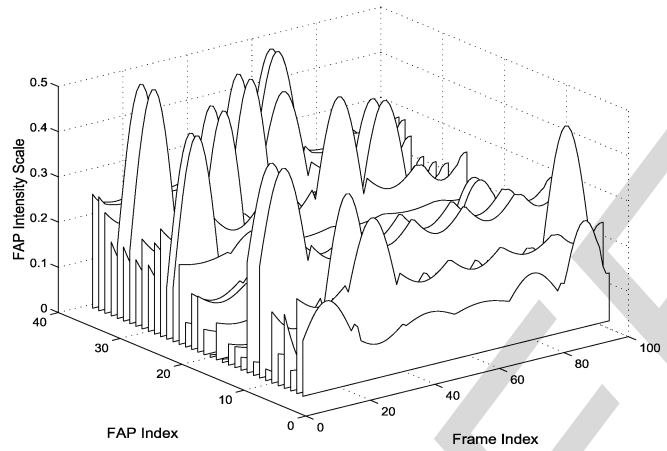

Fig. 18. FAPs are reconstructed at the synthesizer, which shows that the FAP intensity evolves in accordance with the intensity of facial expressions as given in Fig. 17.

figure shows that an expression starts from the neutral and increases its intensity gradually to the apex (the maximum excursion of the facial muscle), then returns gradually to the neutral. Fig. 18 shows the reconstructed FAPs. It shows clearly that the intensity of the reconstructed FAPs evolve smoothly in accordance with the intensity of facial expressions. The FAP intensity provides quantitative information about an evolving facial expression. In other words, the reconstructed FAPs can animate the temporal development of facial expressions. Fig. 19 presents the reconstructed FAPs in accordance with the facial expression analysis result in Fig. 12, which is indeed a noisy version of an ideal curve (see Fig. 18) because of the feature detection errors.

To evaluate the reconstructed FAPs, we compare the reconstructed facial expression at the synthesizer 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 synthesizer is produced by the reconstructed FAPs. For this study, the image sequences are from Cohn-Kanade facial expression database [39]. To give a clear example, the sequence "$S046\_005$" is used. It contains 23 frames and the expression starts from the neutral

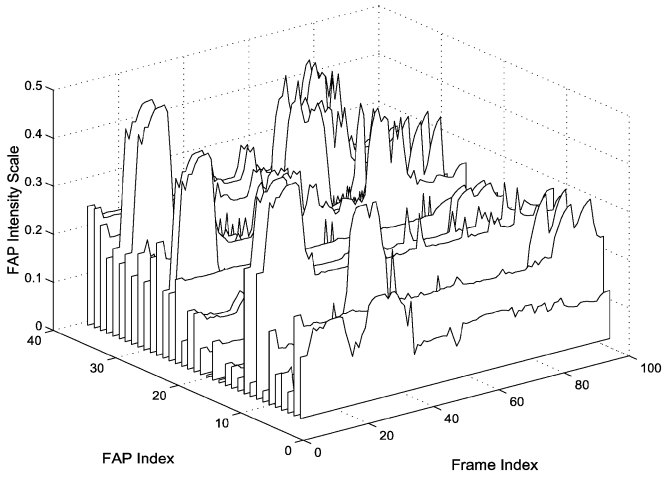

Fig. 19. FAPs are reconstructed at the synthesizer, which shows that the FAP intensity evolves in accordance with the intensity of facial expressions as given in Fig. 12.

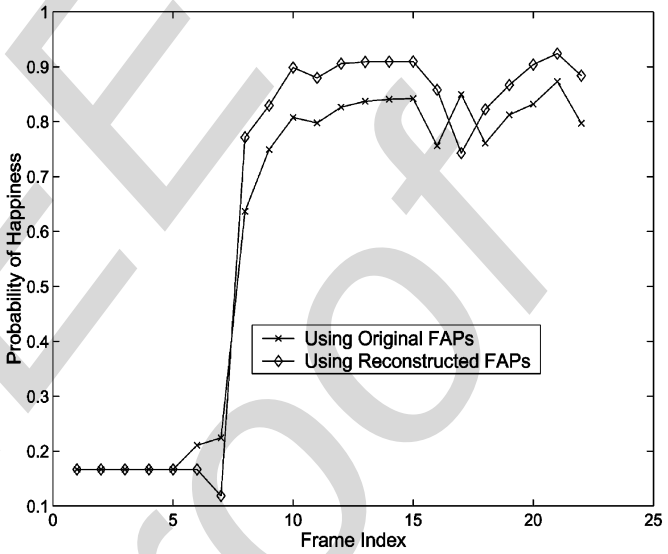

Fig. 20. Comparison between a facial expression generated by the original FAPs and that generated by the reconstructed FAPs. Only the probability of happiness is shown.

to the apex. Fig. 20 shows the two expression intensity curves, where one uses the original detected FAPs at the analysis end and the another uses the reconstructed FAPs at the synthesizer. Because the probabilities of other expressions are negligible, for clarity Fig. 20 plots only the probability of happiness. Though the original FAPs and the reconstructed FAPs could be different, but we can see that the two curves are very close and they track each other well. We tested a number of sequences in the database and the same conclusion can be made. This demonstrates that the reconstructed FAPs are visually faithful to the given expression.

Now we show the robustness of our approach under FAP measurement errors. To create a clear example, we first use the above sequence and manually change FAP amplitudes to simulate the significant measurement errors. To simulate the measurement errors, we manually change FAP values. Specifically, the measurement of FAP4 of frame 13 is changed to 0, and FAP4 of

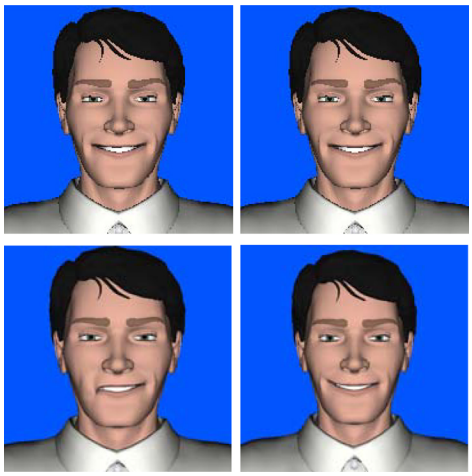

Fig. 21. Animation result of the 13th and 15th frames with FAP errors. Top row: the animation result from our approach, shows good perceptual quality with less influence by FAP errors. Bottom row: the animation result by using directly the original FAPs, shows there are the visible animation artifacts around the mouth corners.

frame 15 is added by half of its value. Fig. 21 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. 21). 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 exist animation artifacts around the left corner of the mouth caused by the FAP measurement errors (the bottom row of Fig. 21). 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 do not cause visible animation artifacts to affect the perceptual quality of the resulting animation.

To further study the performance of our method under random noise, we further selected 18 facial expression sequences (three sequences for each expression) from ten subjects in the Cohn–Kanade database. To minimize the influence by our automated feature detector, the FAPs are first generated by manually marked facial features. Then, for each frame we randomly select four FAPs from a total of 27 FAPs and add a random noise to each selected FAP. The visual inspection of the reconstructed FAPs reveals that our method is able to tolerate the random facial feature errors as well.

VIII. Conclusion

In light of the 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. However, the physiological studies show that temporal course information is necessary for those desiring life-like facial animation [4]. This is the key motivation of this work. 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 benefits.

1) To synthesize six pose-variable facial expressions, our approach needs to transmit only 9 bytes of data per frame to the synthesizer. It is particularly suitable for the interactive facial animation applications, where very low bitrate transmission is required.
2) The temporal course of facial expressions can be synthesized by means of modelling dynamics of facial expressions. This is particularly important for the lifelike facial animation.
3) Unlike streaming directly the FAPs to the synthesizer, the perceptual quality of animation in our approach is less affected by the misdetected facial features.

However, our approach is incapable of reproducing the individuality of a facial expression because the semantic relations between the FAPs and the facial expressions are parameterized by facial muscular actions from psychological studies, and such relations are person-independent. Still for many animation applications, the individuality is not important as the facial model itself is synthetic. In addition, we assume that the receiver end has the required computational power to perform Bayesian inference. Finally, the evaluation of our method is mostly qualitative through visual inspection. A more quantitative performance evaluation of our method is needed. This requires to first establish a quantitative measure to quantify the quality of the reconstructed FAPs and their temporal development. We will pursue this study in the future.

Acknowledgment

The authors gratefully acknowledge the constructive comments from the anonymous reviewers that significantly improved the presentation of this paper.

References

[1] *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*. Englewood Cliffs, NJ: 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 Human Interaction Lab, Univ. of California, San Francisco, 1992, Tech. rep..
[5] Y. Zhang and Q. Ji, "Active and dynamic information fusion for facial expression understanding from image sequence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 699–714, May 2005.
[6] H. Kobayashi and F. Hara, "Facial interaction between animated 3-D face robot and human beings," in *Proc. Int. Conf. Syst., Man, Cybern.*, 1997, pp. 3732–3737.
[7] C. Padgett and G. Cottrell, M. Mozer, M. Jordan, and T. Petsche, Eds., "Representing face images for emotion classification," in *Advances in Neural Information Processing Systems*, 1997, vol. 9***[Author: Is this a book? If so, please provide the publisher.--Ed.]***.
[8] M. J. Lyons, J. Budynek, and S. Akamatsu, "Automatic classification of single facial images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 12, pp. 1357–1362, Dec. 1999.
[9] M. Pantic and L. Rothkrantz, "Expert system for automatic analysis of facial expression," *J. Image Vis. Computing*, vol. 18, no. 11, pp. 881–905, 2000.
[10] Y. Yacoob and L. S. Davis, "Recognizing human facial expressions from long image sequences using optical flow," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 6, pp. 636–642, Jun. 1996.
[11] M. J. Black and Y. Yacoob, "Recognizing facial expression in image sequences using local parameterized models of image motion," *Int. J. Comput. Vis.*, vol. 25, no. 1, pp. 23–48, 1997.

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

[13] N. Oliver, A. Pentland, and F. Bérard, "LAFTER: Lips and face real time tracker with facial expression recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 1997***[Author: Please provide page numbers.--Ed.]***.

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

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

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

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

[18] J. Ahlberg, "An active model for facial feature tracking," *EURASIP J. Appl. Signal Process.*, vol. 2002, no. 6, pp. 566–571, 2002.

[19] 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 Process. Mag.*, vol. 2001, no. 1, pp. 33–80, Jan. 2001.

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

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

[22] P. Eister and B. Girod, "Analyzing facial expressions for virtual conferencing," *IEEE Comput. Graphics Appl.*, pp. 70–79, Sep.–Oct. 1998.

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

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

[25] 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. Circuits Syst. Video Technol.*, vol. 9, no. 2, pp. 277–289, Apr. 1999.

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

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

[28] G. A. Abrantes and F. Pereira, "MPEG-4 facial animation technology: Survey, implementation, and results," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 2, pp. 290–305, Apr. 1999.

[29] 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. Visual. Graphics*, vol. 10, no. 3, pp. 339–352, 2004.

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

[31] K. Waters, "A muscle model for animating three-dimentional facial expression," in *Proc. SIGGRAPH*, 1987***[Author: Please provide page numbers.--Ed.]***.

[32] 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. Circuits Syst. Video Technol.*, vol. 9, no. 2, pp. 264–276, Apr. 1999.

[33] J. Ahlberg and H. Li, "Representation and compressing facial animation parameters using facial action basis functions," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 3, pp. 405–410, Jun. 1999.

[34] P. Wang and Q. Ji, "Learning discriminant feature for multi-view face and eye detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2005.

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

[36] H. Gu, Q. Ji, and Z. Zhu, "Active facial tracking for fatigue detection," in *Proc. IEEE Workshop Applications of Comput. Vis.*, 2002***[Author: Please provide page numbers.--Ed.]***.

[37] Z. Zhu and Q. Ji, "Robust pose invariant facial feature detection and tracking in real-time," in *Proc. Int. Conf. Pattern Recogn.*, 2006***[Author: Please provide page numbers.--Ed.]***.

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

[39] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, 2000, pp. 46–53.

Yongmian Zhang (M'04) received the Ph.D. degree in computer engineering from the University of Nevada, Reno, in 2004.

He held a research position with the Department of Electrical, Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY. His areas of research include information fusion, computer vision, and human–computer interactions.

Qiang Ji (SM'04) received the Ph.D. degree in electrical engineering from the University of Washington, Seattle, in 1998.

He is currently an Associate Professor with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute (RPI), Troy, NY. Prior to joining RPI in 2001, he was an Assistant Professor with the Department of Computer Science, University of Nevada, Reno. He also held research and visiting positions with the Beckman Institute, University of Illinois at Urbana-Champaign, the Robotics Institute, Carnegie Mellon University, and the US Air Force Research Laboratory. His research interests are in computer vision, probabilistic reasoning with Bayesian networks for decision making and information fusion, human–computer interaction, pattern recognition, and robotics. He has published over 100 papers in peer-reviewed journals and conferences. His research has been funded by local and federal government agencies including the National Science Foundation, the National Institutes of Health, the Air Force Office of Scientific Research, the Office of Naval Research, the Defense Advanced Research Projects Agency, the Air Force Research Office and by private companies including Boeing and Honda.

Zhiwei Zhu received the B.S. degree in computer science from the University of Science and Technology, Beijing, China, in 2000 and the Ph.D. degree in electrical engineering from Rensselaer Polytechnic Institute, Troy, NY, in 2005.

He is currently a Member of Technical Staff with the Vision and Robotics Laboratory, Sarnoff Corporation, Princeton, NJ. His research interests include computer vision and pattern recognition.

Beifang Yi received the Ph.D. degree in computer science and engineering from the University of Nevada, Reno.

He is currently a Visiting Assistant Professor with the Department of Computer and Information Sciences, State University of New York, Fredonia. His research interests are in human–computer interactions, graphics, multimedia system, and software design and development.