Int J Comput Vis (2008) 76: 257–281
DOI 10.1007/s11263-007-0059-7

Simultaneous Facial Action Tracking and Expression Recognition in the Presence of Head Motion

Fadi Dornaika · Franck Davoine

Received: 5 October 2006 / Accepted: 12 April 2007 / Published online: 6 July 2007

Abstract The recognition of facial gestures and expressions in image sequences is an important and challenging problem. Most of the existing methods adopt the following paradigm. First, facial actions/features are retrieved from the images, then the facial expression is recognized based on the retrieved temporal parameters. In contrast to this mainstream approach, this paper introduces a new approach allowing the simultaneous retrieval of facial actions and expression using a particle filter adopting multi-class dynamics that are conditioned on the expression. For each frame in the video sequence, our approach is split into two consecutive stages. In the first stage, the 3D head pose is retrieved using a deterministic registration technique based on Online Appearance Models. In the second stage, the facial actions as well as the facial expression are simultaneously retrieved using a stochastic framework based on second-order Markov chains. The proposed fast scheme is either as robust as, or more robust than existing ones in a number of respects. We describe extensive experiments and provide evaluations of performance to show the feasibility and robustness of the proposed approach.

Keywords Simultaneous tracking and recognition · Face and facial feature tracking · Facial expression recognition · Particle filtering

F. Dornaika (✉)
Institut Géographique National, Laboratoire MATIS, 2 avenue Pasteur, 94165 Saint-Mandé Cedex, France
e-mail: fadi.dornaika@ign.fr

F. Davoine
Heudiasyc Mixed Research Unit, CNRS/UTC, 60205 Compiegne, France
e-mail: fdavoine@hds.utc.fr

1 Introduction

The human face has attracted attention in a number of areas including psychology, computer vision, human-computer interaction and computer graphics (Chandrasiri et al. 2004). Human-machine interfaces will require an increasingly good understanding of a subject's behavior so that machines can react accordingly. Facial expression analysis can bring user and computer closer. One challenge is to construct robust, real-time, fully automatic systems to track the facial features and expressions. Many computer vision researchers have been working on tracking and recognition of the whole face or parts of the face. Within the past decade, much work has been done on automatic recognition of facial expression. Computational facial expression analysis is a challenging research topic. The initial 2D methods had limited success mainly because their dependency on the camera viewing angle. One of the main motivations behind 3D methods for face or expression recognition is to enable a broader range of camera viewing angles (Blanz and Vetter 2003; Gokturk et al. 2002; Lu et al. 2006; Moreno et al. 2002; Wang et al. 2004; Wen and Huang 2003; Yilmaz et al. 2002).

To classify expressions in static images many techniques have been proposed, such as those based on neural networks (Tian et al. 2001), Gabor wavelets (Bartlett et al. 2004), and Adaboost (Wang et al. 2004). Recently, more attention has been given to modeling facial deformation in dynamic scenarios, since it is argued that information based on dynamics is richer than that provided by static images. Static image classifiers use feature vectors related to a single frame to perform classification (Lyons et al. 1999). Temporal classifiers try to capture the temporal pattern in the sequence of feature vectors related to each frame. These include the Hidden Markov Model (HMM) based methods (Cohen et al. 2003) and Dynamic Bayesian Networks (DBNs) (Zhang

and Ji 2005). In (Cohen et al. 2003), the authors introduce a facial expression recognition from live video input using temporal cues. They propose a new HMM architecture for automatically segmenting and recognizing human facial expression from video sequences. The architecture performs both segmentation and recognition of the facial expressions automatically using a multi-level architecture composed of an HMM layer and a Markov model layer. In (Zhang and Ji 2005), the authors present a new approach to spontaneous facial expression understanding in image sequences. The facial feature detection and tracking is based on active Infra Red illumination. Modeling dynamic behavior of facial expression in image sequences falls within the framework of information fusion with DBNs.

Surveys of facial expression recognition methods can be found in (Fasel and Luettin 2003; Pantic and Rothkrantz 2000). A number of earlier systems were based on facial motion encoded as a dense flow between successive image frames. However, flow estimates are easily disturbed by illumination changes and non-rigid motion. In (Yacoob and Davis 1996), the authors compute optical flow of regions on the face, then they use a rule-based classifier to recognize the six basic facial expressions. Extracting and tracking facial actions in a video can be done in several ways. In (Bascle and Black 1998), the authors use active contours for tracking the performer's facial deformations. In (Ahlberg 2002), the author retrieves facial actions using a variant of Active Appearance models. The dominant paradigm involves computing a time-varying description of facial actions/features from which the expression can be recognized; that is to say, the tracking process is performed prior to the recognition process (Dornaika and Davoine 2005b; Zhang and Ji 2005). In (Liao and Cohen 2005), authors used a graphical model for modeling the interdependencies of defined facial regions for characterizing facial gestures under varying pose.

However, the results of both processes affect each other in various ways. Since these two problems are interdependent, solving them simultaneously increases reliability and robustness of the results. Such robustness is required when perturbing factors such as partial occlusions, ultra-rapid movements and video streaming discontinuity may affect the input data. Although the idea of merging tracking and recognition is not new, our work addresses two complicated tasks, namely tracking the facial actions and recognizing expression over time in a monocular video sequence.

In the literature, simultaneous tracking and recognition has been used in simple cases. For example, (North et al. 2000) employs a particle-filter-based algorithm for tracking and recognizing the motion class of a juggled ball in 2D. Another example is given in (Zhou et al. 2003); this work proposes a framework allowing the simultaneous tracking and recognizing of human faces using a particle filtering method. The recognition consists in determining a person's identity, which is fixed for the whole probe video. The authors use a mixed state vector formed by the 2D global face motion (affine transform) and an identity variable. However, this work does not address either facial deformation or facial expression recognition.

In this paper, we propose a framework for simultaneous facial action tracking and expression recognition given natural head motion. First, our proposed method estimates the 3D head pose using a deterministic approach based on the principles of Online Appearance Models (OAMs). Second, facial actions and expression are simultaneously estimated using a stochastic approach based on a particle filter adopting mixed states (Isard and Blake 1998). The paper is an extended version of our work (Dornaika and Davoine 2005a). The proposed framework is simple, efficient and robust with respect to head motion given that (1) the dynamic models directly relate the facial actions to the universal expressions, (2) the learning stage does not deal with facial images but only concerns the estimation of auto-regressive models from sequences of facial actions, which is carried out using closed-from solutions, and (3) facial actions are related to a deformable 3D model and not to entities measured in the image plane.

The rest of the paper is organized as follows. Section 2 describes the deformable 3D face model that we use to create shape-free facial patches from input images. Section 3 describes the problem we are focusing on. It presents the adaptive observation model as well as the learning of facial action dynamic models associated with the six universal facial expressions. Section 4 describes the proposed approach, that is, (i) the retrieval of the 3D head pose by a deterministic registration technique, and (ii) the simultaneous retrieval of facial actions and expression using a stochastic framework. Section 5 gives some experimental results and provides a performance evaluation of the developed approach. Section 6 reports results including subject-dependent dynamics. Section 7 concludes the paper.

2 Modeling Faces

2.1 A Deformable 3D Model

In our study, we use the *Candide* 3D face model (Ahlberg 2002). This 3D deformable wireframe model was first developed for the purposes of model-based image coding and computer animation. The 3D shape of this wireframe model (triangular mesh) is directly recorded in coordinate form. It is given by the coordinates of the 3D vertices $\mathbf{P}_i$, $i = 1, \ldots, n$ where n is the number of vertices. Thus, the shape up to a global scale can be fully described by the $3n$-vector

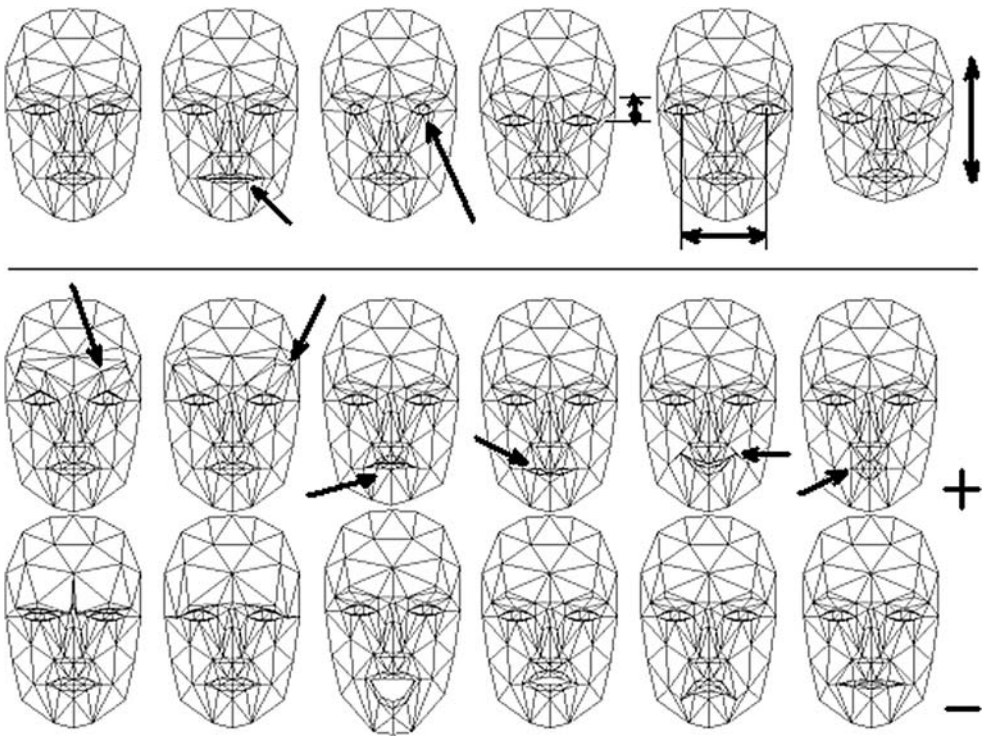


Fig. 1 *First row*: Facial Shape units (neutral shape, mouth width, eyes width, eyes vertical position, eye separation distance, head height). *Second and third rows*: Positive and negative perturbations of Facial Action Units (Brow lowerer, Outer brow raiser, Jaw drop, Upper lip raiser, Lip corner depressor, Lip stretcher)

g; the concatenation of the 3D coordinates of all vertices $\mathbf{P}_i$. The vector **g** is written as:

$$\mathbf{g} = \overline{\mathbf{g}} + \mathbf{S}\boldsymbol{\tau}_\mathbf{s} + \mathbf{A}\boldsymbol{\tau}_\mathbf{a} \tag{1}$$

where $\overline{\mathbf{g}}$ is the standard shape of the model, $\boldsymbol{\tau}_\mathbf{s}$ and $\boldsymbol{\tau}_\mathbf{a}$ are shape and animation control vectors, respectively, and the columns of **S** and **A** are the Shape and Animation Units. A Shape Unit provides a means of deforming the 3D wireframe so as to be able to adapt eye width, head width, eye separation distance, etc. Thus, the term $\mathbf{S}\boldsymbol{\tau}_\mathbf{s}$ accounts for shape variability (inter-person variability) while the term $\mathbf{A}\boldsymbol{\tau}_\mathbf{a}$ accounts for the facial animation (intra-person variability). The shape and animation variabilities can be approximated well enough for practical purposes by this linear relation. Also, we assume that the two kinds of variability are independent. With this model, the ideal neutral face configuration is represented by $\boldsymbol{\tau}_\mathbf{a} = \mathbf{0}$. The shape modes were created manually to accommodate the subjectively most important changes in facial shape (face height/width ratio, horizontal and vertical positions of facial features, eye separation distance). Even though a PCA was initially performed on manually adapted models in order to compute the shape modes, we preferred to consider the *Candide* model with manually created shape modes with semantic signification that are easy to use by human operators who need to adapt the 3D mesh to facial images. The animation modes were measured from pictorial examples in the Facial Action Coding System (FACS) (Ekman and Friesen 1977).

In this study, we use twelve modes for the facial Shape Units matrix **S** and six modes for the facial Animation Units (AUs) matrix **A**. Without loss of generality, we have chosen the six following AUs: lower lip depressor, lip stretcher, lip corner depressor, upper lip raiser, eyebrow lowerer and outer eyebrow raiser. These AUs are enough to cover most common facial animations (mouth and eyebrow movements). Moreover, they are essential for conveying emotions. The effects of the Shape Units and the six Animation Units on the 3D wireframe model are illustrated in Fig. 1.

In (1), the 3D shape is expressed in a local coordinate system. However, one should relate the 3D coordinates to the image coordinate system. To this end, we adopt the weak perspective projection model. We neglect the perspective effects since the depth variation of the face can be considered as small compared to its absolute depth. Therefore, the mapping between the 3D face model and the image is given by a 2×4 matrix, **M**, encapsulating both the 3D head pose and the camera parameters.

Thus, a 3D vertex $\mathbf{P}_i = (X_i, Y_i, Z_i)^T \subset \mathbf{g}$ will be projected onto the image point $\mathbf{p}_i = (u_i, v_i)^T$ given by:

$$(u_i, v_i)^T = \mathbf{M}(X_i, Y_i, Z_i, 1)^T. \tag{2}$$

For a given subject, $\boldsymbol{\tau}_s$ is constant. Estimating $\boldsymbol{\tau}_s$ can be carried out using either feature-based (Lu et al. 2001) or featureless approaches (Ahlberg 2002). In our work, we assume that the control vector $\boldsymbol{\tau}_s$ is already known for every subject, and it is set manually using for instance the face in the first

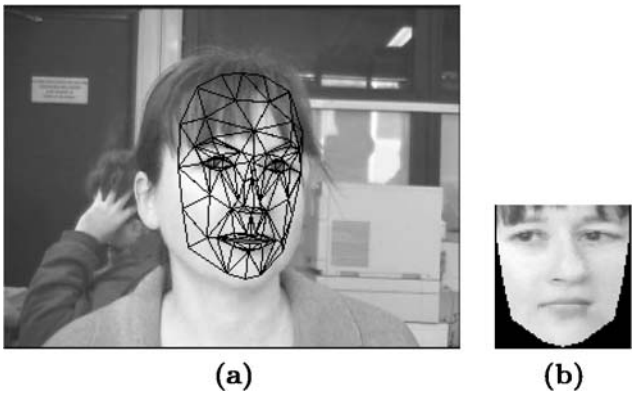


Fig. 2 **a** An input image with correct adaptation of the 3D model. **b** The corresponding shape-free facial image

frame of the video sequence (the *Candide* model and target face shapes are aligned manually). Therefore, (1) becomes:

$$\mathbf{g} = \mathbf{g}_s + \mathbf{A}\boldsymbol{\tau}_a \tag{3}$$

where $\mathbf{g}_s$ represents the static shape of the face—the neutral face configuration. Thus, the state of the 3D wireframe model is given by the 3D head pose parameters (three rotations and three translations) and the animation control vector $\boldsymbol{\tau}_\mathbf{a}$. This is given by the 12-dimensional vector $\mathbf{b}$:

$$\mathbf{b} = [\theta_x, \theta_y, \theta_z, t_x, t_y, t_z, \boldsymbol{\tau}_\mathbf{a}^T]^T \tag{4}$$

$$= [\mathbf{h}^T, \boldsymbol{\tau}_\mathbf{a}^T]^T \tag{5}$$

where the vector $\mathbf{h}$ represents the six degrees of freedom associated with the 3D head pose.

2.2 Shape-Free Facial Patches

A facial patch is represented as a shape-free image (geometrically normalized rawbrightness image). The geometry of this image is obtained by projecting the standard shape $\overline{\mathbf{g}}$ with a centered frontal 3D pose onto an image with a given resolution. The geometrically normalized image is obtained by texture mapping from the triangular 2D mesh in the input image (see Fig. 2) using a piece-wise affine transform, $\mathcal{W}$. The warping process applied to an input image $\mathbf{y}$ is denoted by:

$$\mathbf{x}(\mathbf{b}) = \mathcal{W}(\mathbf{y}, \mathbf{b}) \tag{6}$$

where $\mathbf{x}$ denotes the shape-free patch and $\mathbf{b}$ denotes the geometrical parameters. Several resolution levels can be chosen for the shape-free patches. The reported results are obtained with a shape-free patch of 5392 pixels. Regarding photometric transformations, a zero-mean unit-variance normalization is used to partially compensate for contrast variations. The complete image transformation is implemented as follows: (i) transfer the rawbrightness facial patch $\mathbf{y}$ using the piece-wise affine transform associated with the vector $\mathbf{b}$, and (ii) perform the gray-level normalization of the obtained patch.

3 Background and Problem Formulation

Given a video sequence depicting a moving head/face, we would like to recover, for each frame, the 3D head pose, the facial actions encoded by the control vector $\boldsymbol{\tau}_\mathbf{a}$ as well as the facial expression. In other words, we would like to estimate the vector $\mathbf{b}_t$ (see (5)) at time t in addition to the facial expression given all the observed data up to time t, denoted $\mathbf{y}_{1:t} \equiv \{\mathbf{y}_1, \ldots, \mathbf{y}_t\}$. In a tracking context, the model parameters associated with the current frame will be carried over to the next frame. Since the facial expression can be considered as a random discrete variable, we need to append to the continuous state vector $\mathbf{b}_t$ a discrete state component γ_t in order to create a mixed state:

$$\begin{pmatrix} \mathbf{b}_t \\ \gamma_t \end{pmatrix} \tag{7}$$

where $\gamma_t \in \mathcal{E} = \{1, 2, \ldots, N_\gamma\}$ is the discrete component of the state, drawn from a finite set of integer labels. Each integer label represents one of the six universal expressions: surprise, disgust, fear, joy, sadness and anger. In our study, we adopt these facial expressions together with the neutral expression, that is, N_γ is set to 7. There is another useful representation of the mixed state which is given by:

$$\begin{pmatrix} \mathbf{h}_t \\ \mathbf{a}_t \end{pmatrix} \tag{8}$$

where $\mathbf{h}_t$ denotes the 3D head pose parameters, and $\mathbf{a}_t$ the facial actions appended with the expression label γ_t, i.e. $\mathbf{a}_t = [\boldsymbol{\tau}_{\mathbf{a}_{(t)}}^T, \gamma_t]^T$.

This separation is consistent with the fact that the facial expression is highly correlated with the facial actions, while the 3D head pose is independent of the facial actions and expressions.

3.1 Adaptive Observation Model

For each input frame $\mathbf{y}_t$, the observation is simply the warped (shape-free) facial patch associated with the geometric parameters $\mathbf{b}_t$. We use the HAT symbol for the tracked parameters and images. For a given frame t, $\hat{\mathbf{b}}_t$ represents the computed geometric parameters and $\hat{\mathbf{x}}_t$ the corresponding shape-free patch, that is,

$$\hat{\mathbf{x}}_t = \mathbf{x}(\hat{\mathbf{b}}_t) = \mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_t). \tag{9}$$

The estimation of $\hat{\mathbf{b}}_t$ from the sequence of images will be presented in Sect. 4. $\hat{\mathbf{b}}_0$ is initialized manually, according to the face in the first video frame.

The appearance model associated with the shape-free facial patch at time t, $\mathcal{A}_t$, is time-varying in that it models the appearances present in all observations $\hat{\mathbf{x}}$ up to time $t-1$. This may be required as a result, for instance, of illumination changes or out-of-plane rotated faces.

By assuming that the pixels within the shape-free patch are independent, we can model the appearance using a multivariate Gaussian with a diagonal covariance matrix $\mathbf{\Sigma}$. Although the independence assumption may be violated, at least locally, we adopt it in our work in order to keep the problem tractable. The choice of a Gaussian distribution is motivated by the fact that this kind of distribution provides a simple and general model for additive noises. In other words, this multivariate Gaussian is the distribution of the facial patches $\hat{\mathbf{x}}_t$. Let $\boldsymbol{\mu}$ be the Gaussian center and $\boldsymbol{\sigma}$ the vector containing the square root of the diagonal elements of the covariance matrix $\mathbf{\Sigma}$. $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are d-vectors (d is the size of $\mathbf{x}$).

In summary, the observation likelihood is written as:

$$p(\mathbf{y}_t|\mathbf{b}_t) = p(\mathbf{x}_t|\mathbf{b}_t) = \prod_{i=1}^{d} \mathbf{N}(x_i; \mu_i, \sigma_i)_t \tag{10}$$

where $\mathbf{N}(x_i; \mu_i, \sigma_i)$ is the normal density:

$$\mathbf{N}(x_i; \mu_i, \sigma_i) = (2\pi\sigma_i^2)^{-1/2} \exp\left[-\frac{1}{2}\left(\frac{x_i - \mu_i}{\sigma_i}\right)^2\right]. \tag{11}$$

We assume that the appearance model $\mathcal{A}_t$ summarizes the past observations under an exponential envelope with a forgetting factor $\alpha = 1 - \exp(-\frac{\log 2}{n_h})$, where n_h represents the half-life of the envelope in frames (Jepson et al. 2003).

When the patch $\hat{\mathbf{x}}_t$ is available at time t, the appearance is updated and used to track in the next frame. It can be shown that the appearance model parameters, i.e., the μ_i's and σ_i's can be updated from time t to time $(t+1)$ using the following equations (see Jepson et al. 2003 for more details on OAMs):

$$\mu_{i_{(t+1)}} = (1-\alpha)\mu_{i_{(t)}} + \alpha\hat{x}_{i_{(t)}}, \tag{12}$$

$$\sigma^2_{i_{(t+1)}} = (1-\alpha)\sigma^2_{i_{(t)}} + \alpha(\hat{x}_{i_{(t)}} - \mu_{i_{(t)}})^2. \tag{13}$$

This technique is simple, time-efficient and therefore suitable for real-time applications. The appearance parameters reflect the most recent observations within a roughly $L = 1/\alpha$ window with exponential decay. Figure 3 shows an envelope having α equal to 0.01 where the current frame is 500.

Note that $\boldsymbol{\mu}$ is initialized with the first patch $\hat{\mathbf{x}}_0$. However, (13) is not used with α being a constant until the number of frames reaches a given value (*e.g.*, the first 40 frames). For these frames, the classical variance is used, that is, (13) is used with α being set to $\frac{1}{t}$.

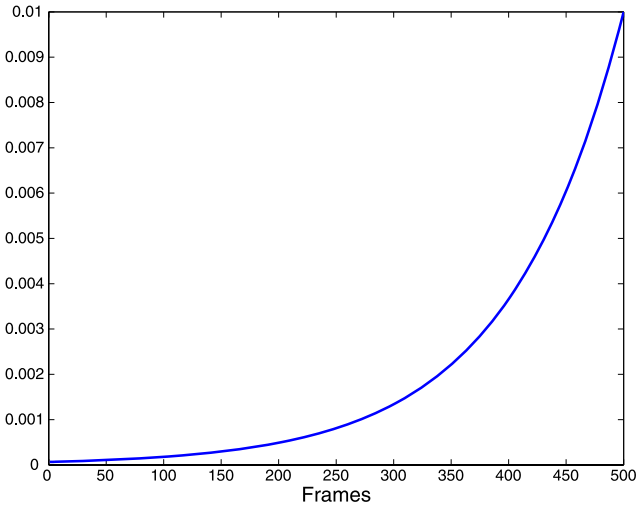


Fig. 3 A sliding exponential envelope with $\alpha = 0.01$. The current frame time is 500

Here we used a single Gaussian to model the appearance of each pixel in the shape-free template. However, modeling the appearance with Gaussian mixtures can also be used at the expense of an additional computational load (e.g., see Lee 2005; Zhou et al. 2004).

3.2 Facial Action Dynamic Models

Corresponding to each basic expression class, γ, there is a stochastic dynamic model describing the temporal evolution of the facial actions $\boldsymbol{\tau}_{\mathbf{a}(t)}$, given the expression. It is assumed to be a Markov model of order K. For each basic expression γ, we associate a Gaussian Auto-Regressive Process defined by:

$$\boldsymbol{\tau}_{\mathbf{a}(t)} = \sum_{k=1}^{K} \mathbf{A}_k^{\gamma} \boldsymbol{\tau}_{\mathbf{a}(t-k)} + \mathbf{d}^{\gamma} + \mathbf{B}^{\gamma}\mathbf{w}_t \tag{14}$$

in which $\mathbf{w}_t$ is a vector of 6 independent random $\mathcal{N}(0,1)$ variables. The parameters of the dynamic model are: (i) deterministic parameters $\mathbf{A}_1^{\gamma}, \mathbf{A}_2^{\gamma}, \ldots, \mathbf{A}_K^{\gamma}$ and $\mathbf{d}^{\gamma}$, and stochastic parameters $\mathbf{B}^{\gamma}$ which are multipliers for the stochastic process $\mathbf{w}_t$. It is worth noting that the above model can be used in predicting the process from the previous K values. The predicted value at time t obeys a multivariate Gaussian centered at the deterministic value of (14), with $\mathbf{B}^{\gamma}\mathbf{B}^{\gamma T}$ being its covariance matrix. In our study, we are interested in second-order models, i.e. $K = 2$. The reason is twofold. First, these models are easy to estimate. Second, they are able to model complex dynamics. For example, these models have been used in (Blake and Isard 2000) for learning the 2D motion of talking lips (profile contours), beating heart, and writing fingers.

3.2.1 Learning the Second-Order Auto-Regressive Models

Given a training sequence $\boldsymbol{\tau}_{\mathbf{a}(1)}, \ldots, \boldsymbol{\tau}_{\mathbf{a}(T)}$, with $T > 2$, belonging to the same expression, it is well known that a Maximum Likelihood Estimator provides a closed-form solution for the model parameters (Blake and Isard 2000). For a second-order model, these parameters reduce to two 6×6 matrices $\mathbf{A}_1^{\gamma}, \mathbf{A}_2^{\gamma}$, a 6-vector $\mathbf{d}^{\gamma}$, and a 6×6 covariance matrix $\mathbf{C}^{\gamma} = \mathbf{B}^{\gamma}\mathbf{B}^{\gamma T}$. Therefore, (14) reduces to:

$$\boldsymbol{\tau}_{\mathbf{a}(t)} = \mathbf{A}_2^{\gamma}\boldsymbol{\tau}_{\mathbf{a}(t-2)} + \mathbf{A}_1^{\gamma}\boldsymbol{\tau}_{\mathbf{a}(t-1)} + \mathbf{d}^{\gamma} + \mathbf{B}^{\gamma}\mathbf{w}_t. \tag{15}$$

The parameters of each auto-regressive model can be computed from temporal facial action sequences (see Appendix). Ideally, the temporal sequence should contain several instances of the corresponding expression.

More details about auto-regressive models and their computation can be found in (Blake and Isard 2000; Ljung 1987; North et al. 2000). It is worth noting that each universal expression has its own second-order auto-regressive model given by (15). However, the dynamics of facial actions associated with the neutral expression can be simpler and are given by:

$$\boldsymbol{\tau}_{\mathbf{a}(t)} = \boldsymbol{\tau}_{\mathbf{a}(t-1)} + \mathbf{D}\mathbf{w}_t$$

where $\mathbf{D}$ is a diagonal matrix whose elements represent the variances around the ideal neutral configuration $\boldsymbol{\tau}_{\mathbf{a}} = \mathbf{0}$. The right-hand side of the above equation is constrained to belong to a predefined interval, since a neutral configuration and expression is characterized by both the lack of motion and the closeness to the ideal static configuration.

3.2.2 Computed Auto-Regressive Models

In our study, the auto-regressive models are learned using a supervised learning scheme. First, we asked a volunteer student to perform each basic expression several times in approximately 30-second sequences. Each video sequence contains several cycles depicting a particular facial expression: Surprise, Sadness, Joy, Disgust, Anger, and Fear. Second, for each training video, the 3D head pose and the facial actions $\boldsymbol{\tau}_{\mathbf{a}(t)}$ are tracked using our deterministic appearance-based tracker (Dornaika and Davoine 2006).

Figure 4 illustrates the value of the facial actions, $\boldsymbol{\tau}_{\mathbf{a}(t)}$, associated with six training video sequences. For clarity purposes, only two components are shown for a given plot. For a given training video, the neutral frames are skipped from the original training sequence used in the computation of the auto-regressive models. Recall that for an ideal neutral configuration for the 3D wireframe, the vector $\boldsymbol{\tau}_{\mathbf{a}(t)}$ is zero.

In order to assess the quality of the auto-regressive models, we have used them for synthesizing facial actions that simulate the basic expressions. To this end, the recursive equation (15) has been used with non-zero initial conditions. Figure 5 shows the synthesized facial actions, $\boldsymbol{\tau}_{\mathbf{a}(t)}$, using the six auto-regressive models constructed using the data of Fig. 4. Each synthesized sequence contains ten cycles associated with one basic expression. Each cycle consists of 30 generated samples/frames followed by 30 zero samples (neutral frames). For a given plot, only two components are displayed. As can be seen, the dynamics of the synthesized facial actions are highly consistent with the original training data.

3.3 The Transition Matrix

In our study, the facial actions as well as the expression are simultaneously retrieved using a stochastic framework, namely the particle filtering method. This framework requires a transition matrix $\mathbf{T}$ whose entries $T_{\gamma',\gamma}$ describe the probability of transition between two expression labels γ' and γ. The transition probabilities need to be learned from training video sequences. In the literature, the transition probabilities associated with states (not necessarily facial expressions) are inferred using supervised and unsupervised learning techniques. However, since we are dealing with high level states (the universal facial expressions), we have found that a realistic *a priori* setting works very well. We adopt a 7×7 symmetric matrix whose diagonal elements are close to one (e.g. $T_{\gamma,\gamma} = 0.8$, that is, 80% of the transitions occur within the same expression class). The rest of the percentage is distributed equally among the expressions. In this model, transitions from one expression to another expression without going through the neutral one are allowed. Furthermore, this model adopts the most general case where all universal expressions have the same probability. However, according to the context of the application, one can adopt other transition matrices in which some expressions are more likely to happen than others.

4 Approach

Since at any given time, the 3D head pose parameters can be considered as independent of the facial actions and expression, our basic idea is to split the estimation of the unknown parameters into two main stages. For each input video frame $\mathbf{y}_t$, these two stages are invoked in sequence in order to recover the mixed state $[\mathbf{h}_t^T, \mathbf{a}_t^T]^T$. Our proposed approach is illustrated in Fig. 6. In the first stage, the six degrees of freedom associated with the 3D head pose (encoded by the vector $\mathbf{h}_t$) are obtained using a deterministic registration technique similar to that proposed in (Dornaika and Davoine 2006). In the second stage, the facial actions and the facial expression (encoded by the vector $\mathbf{a}_t = [\boldsymbol{\tau}_{\mathbf{a}(t)}^T, \gamma_t]^T$) are simultaneously estimated using a stochastic framework based on a particle filter. Such models

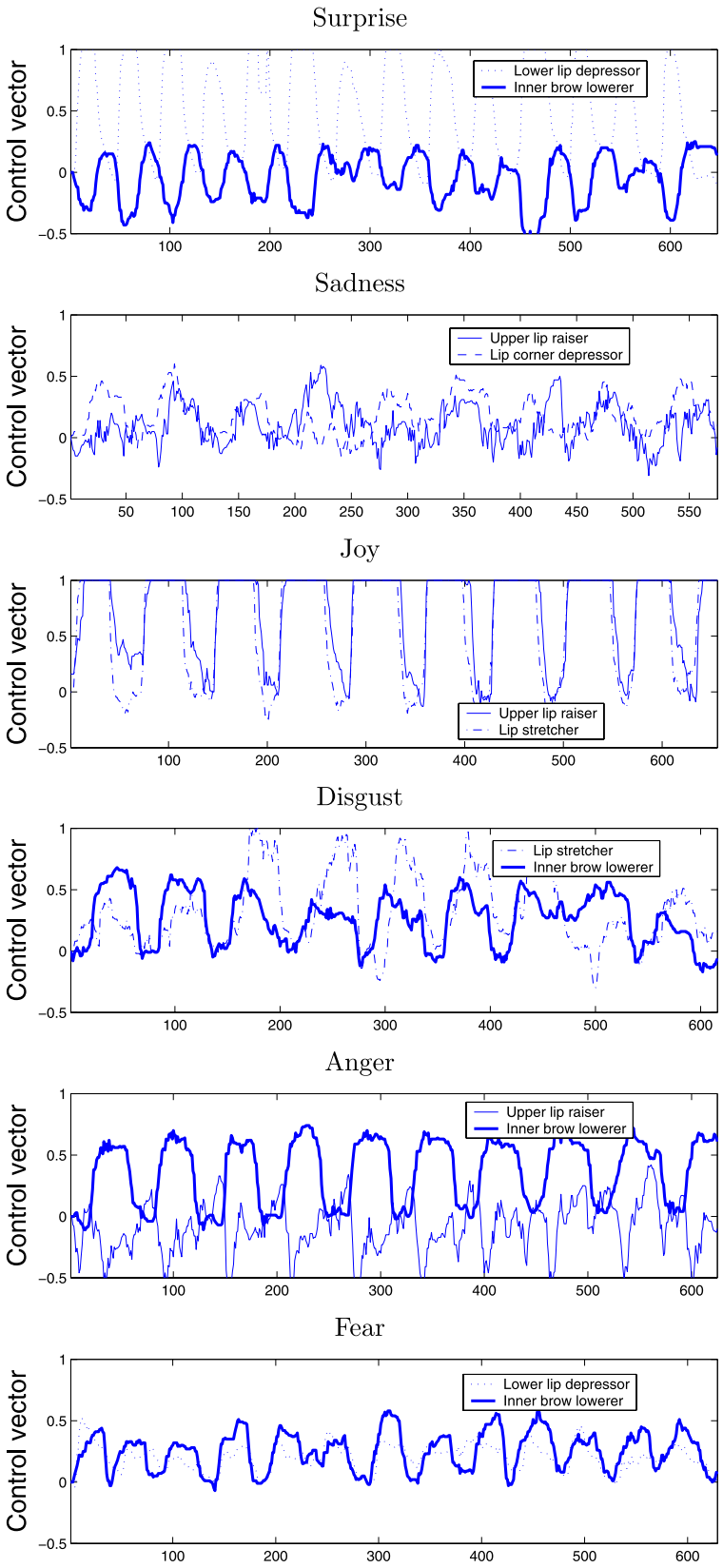


Fig. 4 The automatically tracked facial actions, $\boldsymbol{\tau}_{\mathbf{a}(t)}$, using the training videos. Each video sequence corresponds to one expression. For a given plot, only two components are displayed

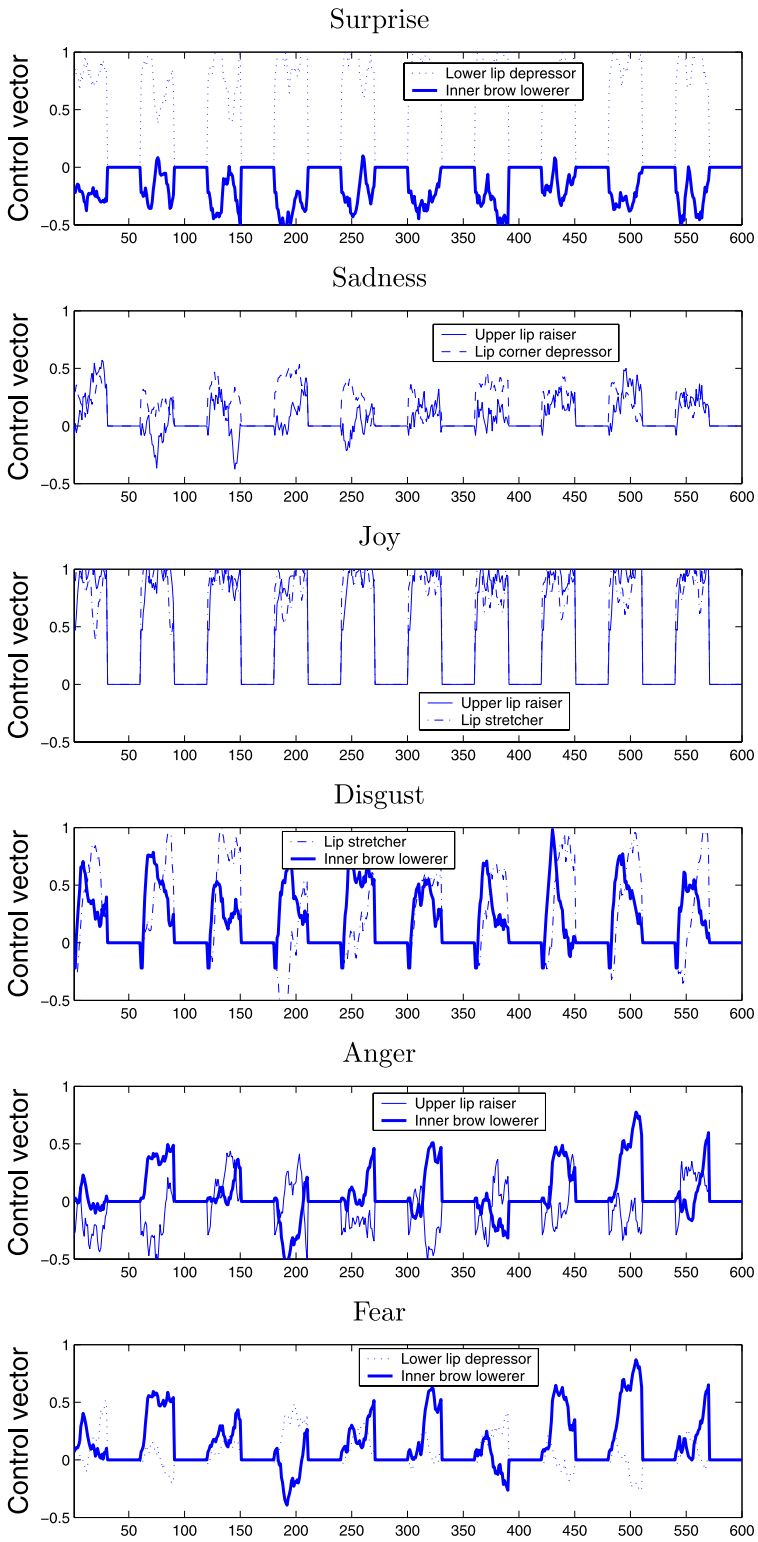


Fig. 5 The synthesized facial actions, $\boldsymbol{\tau}_{\mathbf{a}(t)}$, using the auto-regressive models built with the data of Fig. 4. Each synthesized sequence contains ten cycles associated with one basic facial expression. For a given plot, only two components are displayed

have been used to track objects when different types of dynamics can occur (Isard and Blake 1998). Other examples of auxiliary discrete variables beside the main hidden state of interest are given in (Perez and Vermaak 2005). Since $\boldsymbol{\tau}_{\mathbf{a}(t)}$ and γ_t are highly correlated their simultaneous estimation will give results that are more robust and accurate than results obtained with methods estimating them in sequence. In the following, we present the parameter estimation process associated with the current frame $\mathbf{y}_t$. Recall that the head pose is computed using a deterministic approach, while the

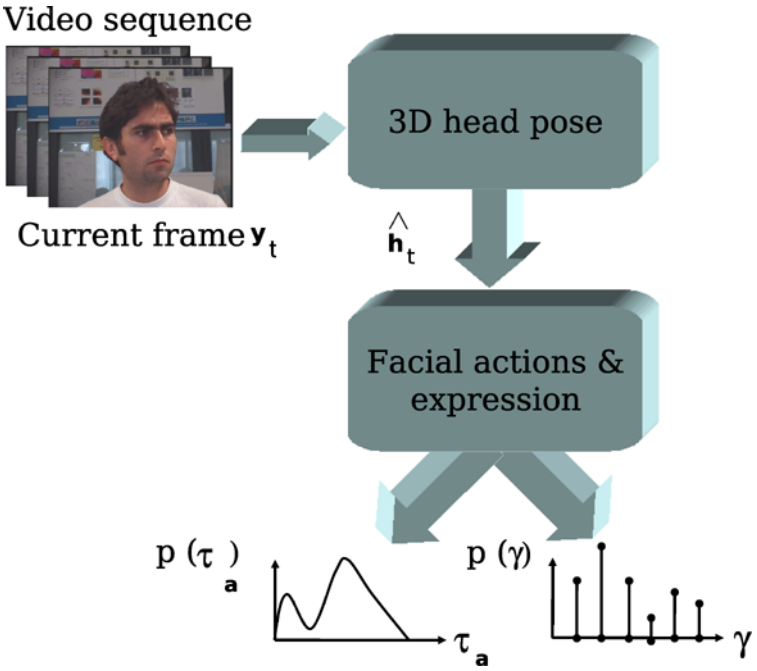


Fig. 6 The proposed two-stage approach. In the first stage (Sect. 4.1), the 3D head pose is computed using a deterministic registration technique. In the second stage (Sect. 4.2), the facial actions and expression are simultaneously estimated using a stochastic technique involving multi-class dynamics

facial actions and expressions are estimated using a probabilistic framework.

4.1 3D Head Pose

The purpose of this stage is to estimate the six degrees of freedom associated with the 3D head pose at frame t, that is, the vector $\mathbf{h}_t$. Our basic idea is to recover the current 3D head pose parameters from the previous 12-vector $\hat{\mathbf{b}}_{t-1} = [\hat{\theta}_{x(t-1)}, \hat{\theta}_{y(t-1)}, \hat{\theta}_{z(t-1)}, \hat{t}_{x(t-1)}, \hat{t}_{y(t-1)}, \hat{t}_{z(t-1)}, \hat{\boldsymbol{\tau}}_{\mathbf{a}(t-1)}]^T = [\hat{\mathbf{h}}_{t-1}, \hat{\boldsymbol{\tau}}_{\mathbf{a}(t-1)}]^T$ using a region-based registration technique. In other words, the current input image $\mathbf{y}_t$ is registered with the current appearance model $\mathcal{A}_t$. For this purpose, we minimize the *Mahalanobis* distance between the warped image patch and the current appearance mean—the current Gaussian center

$$\min_{\mathbf{h}} e(\mathbf{h}_t) = \min d[\mathbf{x}(\mathbf{b}_t), \boldsymbol{\mu}_t] = \min \sum_{i=1}^{d} \left(\frac{x_i - \mu_i}{\sigma_i} \right)^2_{(t)}. \quad (16)$$

The above criterion can be minimized using an iterative gradient descent method where the starting solution is set to the previous solution $\hat{\mathbf{h}}_{t-1}$. The appearance parameters, i.e. the vectors $\boldsymbol{\mu}_t$ and $\boldsymbol{\sigma}_t$, are known using the recursive equations (12) and (13). During the above optimization process the facial actions are set to the constant values $\hat{\boldsymbol{\tau}}_{\mathbf{a}(t-1)}$. Handling outlier pixels (caused for instance by occlusions) is performed by replacing the quadratic function by the Huber's cost function (Dornaika and Davoine 2006; Huber 1981).

Computation of the Gradient Matrix The gradient matrix associated with the 3D head pose parameters is $\mathbf{G} = \frac{\partial \mathcal{W}(\mathbf{y}_t, \mathbf{b}_t)}{\partial \mathbf{h}} = \frac{\partial \mathbf{x}_t}{\partial \mathbf{h}}$. It is approximated by numerical differences.

Once the solution given by the 12-vector $\hat{\mathbf{b}}_t = [\hat{\mathbf{h}}_t^T, \hat{\boldsymbol{\tau}}_{\mathbf{a}(t)}^T]^T$ becomes available for a given frame, it is possible to compute the current gradient matrix from the associated input image. We use the following:

The j^{th} column of $\mathbf{G}$ ($j = 1, \ldots, \dim(\mathbf{h}) = 6$) where $\mathbf{G}_j = \frac{\partial \mathcal{W}(\mathbf{y}_t, \mathbf{b}_t)}{\partial h_j}$ can be estimated using differences

$$\mathbf{G}_j \simeq \frac{\mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_t) - \mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_t + \delta_j \mathbf{q}_j)}{\delta_j}$$

where δ_j is a suitable step size and $\mathbf{q}_j$ is a 12-vector with all elements zero except the j^{th} element, which is equal to one. To gain more accuracy, the j^{th} column of $\mathbf{G}$ is estimated using several steps around the current value b_j. Averaging over all these, we then obtain the final $\mathbf{G}_j$ as

$$\mathbf{G}_j = \frac{1}{K} \sum_{k=-K/2, k\neq 0}^{K/2} \frac{\mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_t) - \mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_t + k\,\delta_j\,\mathbf{q}_j)}{k\delta_j}$$

where δ_j is the smallest perturbation associated with the parameter h_j and K is the number of steps (in our experiments, K is set to 8). One can also use a weighted average that downweights the contribution of perturbations which are far from the current values, i.e., the averaging weights can be taken from triangular or Gaussian windows.

4.2 Simultaneous Facial Actions and Expression

In this stage, our goal is to simultaneously infer the facial actions as well as the expression label associated with the current frame t given (i) the observation model (see (10)), (ii) the dynamics associated with each expression (see (15)), and (iii) the 3D head pose for the current frame computed by the deterministic approach (see Sect. 4.1). This will be performed using a particle filter paradigm. Thus, the statistical inference of such paradigm will provide a posterior distribution for the facial actions $\boldsymbol{\tau}_{\mathbf{a}(t)}$ as well as a Probability Mass function for the facial expression γ_t.

Since the 3D head pose $\mathbf{h}_t$ is already computed, we are left with the mixed state $\mathbf{a}_t = [\boldsymbol{\tau}_{\mathbf{a}(t)}^T, \gamma_t]^T$. The dimension of the vector $\mathbf{a}_t$ is 7. Here we will employ a particle filter algorithm allowing the recursive estimation of the posterior distribution $p(\mathbf{a}_t|\mathbf{x}_{1:(t)})$ using a particle set. This is approximated by a set of J particles $\{(\mathbf{a}_t^{(0)}, w_t^{(0)}), \ldots, (\mathbf{a}_t^{(J)}, w_t^{(J)})\}$. Once this distribution is known the facial actions as well as the expression can be inferred using some loss function such as the MAP or the mean. Figure 7 illustrates the proposed two-stage approach. It shows how the current posterior $p(\mathbf{a}_t|\mathbf{x}_{1:(t)})$ can be inferred from the previous posterior $p(\mathbf{a}_{t-1}|\mathbf{x}_{1:(t-1)})$ using a particle filter algorithm.

On a 3.2 GHz PC, a C code of the approach computes the 3D head pose parameters in 25 ms and the facial actions/expression in 31 ms where the patch resolution is 1310 pixels and the number of particles is 100.

5 Experimental results

In this section, we first report results on simultaneous facial action tracking and expression recognition. Then we present performance studies, considering different perturbing factors such as robustness to rapid facial movements or to imprecise 3D head pose estimation.

5.1 Simultaneous tracking and recognition

Figure 8 shows the application of the proposed approach to a 748-frame test video sequence. The upper part of this figure shows 9 frames of this sequence: 50, 130, 221, 300, 371, 450, 500, 620, and 740. The two plots illustrate the probability of each expression as a function of time (frames). The lower part of this figure shows the tracking results associated with frames 130, 371, and 450. The upper left corner of these frames depicts the appearance mean and the current shape-free facial patch.

Figure 9(a) illustrates the weighted average of the tracked facial actions, $\hat{\boldsymbol{\tau}}_{\mathbf{a}(t)}$. For the sake of clarity, only three out of six components are shown. For this sequence, the maximum probability was correctly indicating the displayed expression. We noticed that some displayed expressions can, during a short initial phase (very few frames), be considered as a mixture of two expressions (the displayed one and another one). This is due to the fact that face postures and dynamics at some transition phases can be shared by more than one expression. This is not a problem since the frame-wise expression probabilities can be merged and averaged over a temporal patch including contiguous non-neutral frames. Figure 9(b) illustrates this scheme and shows the resulting segmentation of the used test video. One remarks that this holds true for a human observer, who may fail to recognize a gesture from only one single frame.

In the above experiment, the total number of particles is set to 200. Figure 10 illustrates the same facial actions when the number of particles is set to 100. We have found that there is no significant difference in the estimated facial actions and expressions when the tracking is performed with 100 particles (see Figs. 9(a) and 10), which is due to the use of learned multi-class dynamics.

Figure 11 shows the tracking results associated with another 600-frame test video sequence depicting significant out-of-plane head movements. The recognition results were correct. Recall that the facial actions are related to the deformable 3D model and thus the recognition based on them is independent from the viewing angle.

A Challenging Example We have dealt with a challenging test video. For this 1600-frame test video, we asked our subject to adopt arbitrarily different facial gestures and expressions for an arbitrary duration and in an arbitrary order. Figure 12 (top) illustrates the probability mass distribution as a function of time. As can be seen, surprise, joy, anger, disgust, and fear are clearly and correctly detected. Also, we find that the facial actions associated with the subject's conversation are correctly tracked using the dynamics of the universal expressions. The tracked facial actions associated with the subject's conversation are depicted in nine frames (see the lower part of Fig. 12). The whole video can be found at http://www.hds.utc.fr/~fdavoine/MovieTrackingRecognition.wmv.

5.2 Performance study

One-Class Dynamics Versus Multi-Class Dynamics In order to show the advantage of using multi-class dynamics and mixed states, we conducted the following experiment. We used a particle filter for tracking facial actions. However, this time the state consists only of facial actions and the dynamics are replaced with a simple noise model, i.e. motion is modeled by a random noise. Figures 13(a) and 13(b) show the tracking results associated with the same input frame. (a) Displays the tracking results obtained with a particle filter adopting a single-class dynamics. (b) Displays the tracking results with our proposed approach adopting the six auto-regressive models. As can be seen, by using

1. **Initialization** $t = 0$:
   - Initialize the 3D head pose $\hat{\mathbf{h}}_0$
   - Generate J state samples $\mathbf{a}_0^{(1)}, \ldots, \mathbf{a}_0^{(J)}$ according to some prior density $p(\mathbf{a}_0)$ and assign them identical weights, $w_0^{(1)} = \ldots = w_0^{(J)} = 1/J$
2. **Tracking** At time step $t \leftarrow t+1$, get the input frame $\mathbf{y}_t$. Compute the corresponding 3D head pose, $\hat{\mathbf{h}}_t$, using the deterministic method outlined in Section 4.1. We have J weighted particles $(\mathbf{a}_{t-1}^{(j)}, w_{t-1}^{(j)})$ that approximate the posterior distribution of the state $p(\mathbf{a}_{t-1}|\mathbf{x}_{1:(t-1)})$ at the previous time step
   (a) Resample the particles proportionally to their weights, *i.e.* particles with high weights are duplicated and particles with small weights are removed. Resampled particles have the same weights
   (b) Draw J particles $\mathbf{a}_t^{(j)}$ according to the dynamic model $p(\mathbf{a}_t|\mathbf{a}_{t-1} = \mathbf{a}_{t-1}^{(j)})$. The obtained new particles approximate the predicted distribution $p(\mathbf{a}_t|\mathbf{x}_{1:(t-1)})$. For multi-class dynamics and mixed states this is done in two steps
   Discrete: Draw an expression label $\gamma_t^{(j)} = \gamma \in \mathcal{E} = \{1, 2, \ldots, N_\gamma\}$ with probability $T_{\gamma',\gamma}$, where $\gamma' = \gamma_{t-1}^{(j)}$
   Continuous: Compute $\boldsymbol{\tau}_{\mathbf{a}(t)}^{(j)}$ as
   $$\boldsymbol{\tau}_{\mathbf{a}(t)}^{(j)} = \mathbf{A}_2^\gamma \boldsymbol{\tau}_{\mathbf{a}(t-2)}^{(j)} + \mathbf{A}_1^\gamma \boldsymbol{\tau}_{\mathbf{a}(t-1)}^{(j)} + \mathbf{d}^\gamma + \mathbf{B}^\gamma \mathbf{w}_t^{(j)}$$
   where $\gamma = \gamma_t^{(j)}$ and $\mathbf{w}_t^{(j)}$ is a 6-vector of standard normal random variables
   (c) Compute the shape-free patch $\mathbf{x}(\mathbf{b}_t^{(j)})$ according to (6) where $\mathbf{b}_t^{(j)} = [\ \hat{\mathbf{h}}_t^T, \quad \boldsymbol{\tau}_{\mathbf{a}(t)}^{(j)T}\ ]^T$
   (d) Weight each new particle proportionally to its likelihood
   $$w_t^{(j)} = \frac{p(\mathbf{x}_t|\mathbf{b}_t^{(j)})}{\sum_{m=1}^{J} p(\mathbf{x}_t|\mathbf{b}_t^{(m)})}$$
   The set of weighted particles approximates the posterior $p(\mathbf{a}_t|\mathbf{x}_{1:t})$
   (e) Set the probability of each basic expression $\gamma^\star \in \mathcal{E} = \{1, 2, \ldots, N_\gamma\}$ to
   $$P(\gamma^\star) = \sum_{m=1}^{J} \begin{cases} w_t^{(m)} & \text{if } \gamma_t^{(m)} = \gamma^\star \\ 0 & \text{otherwise} \end{cases}$$
   (f) Set the facial actions to $\hat{\boldsymbol{\tau}}_{\mathbf{a}(t)} = \sum_{m=1}^{J} w_t^{(m)} \boldsymbol{\tau}_{\mathbf{a}(t)}^{(m)}$
   (g) Set the geometrical parameters as $\hat{\mathbf{b}}_t = [\hat{\mathbf{h}}_t^T, \quad \hat{\boldsymbol{\tau}}_{\mathbf{a}(t)}^T\ ]^T$
   (h) Based on $\hat{\mathbf{b}}_t$, update the appearance (using Eqs. (9), (12), and (13)) as well as the 3D pose gradient matrix (Section 4.1). Go to 2.

Fig. 7 Inferring the 3D head pose, the facial actions and expression. A particle-filter-based algorithm is used for the simultaneous recovery of the facial actions and expression

mixed states with learned multi-class dynamics, the facial action tracking becomes considerably more accurate (see the adaptation of the mouth region—the lower lip).

Effect of Rapid and/or Discontinuous Facial Movements It is well known that facial expressions introduce rapid facial feature movements, and hence many developed track-

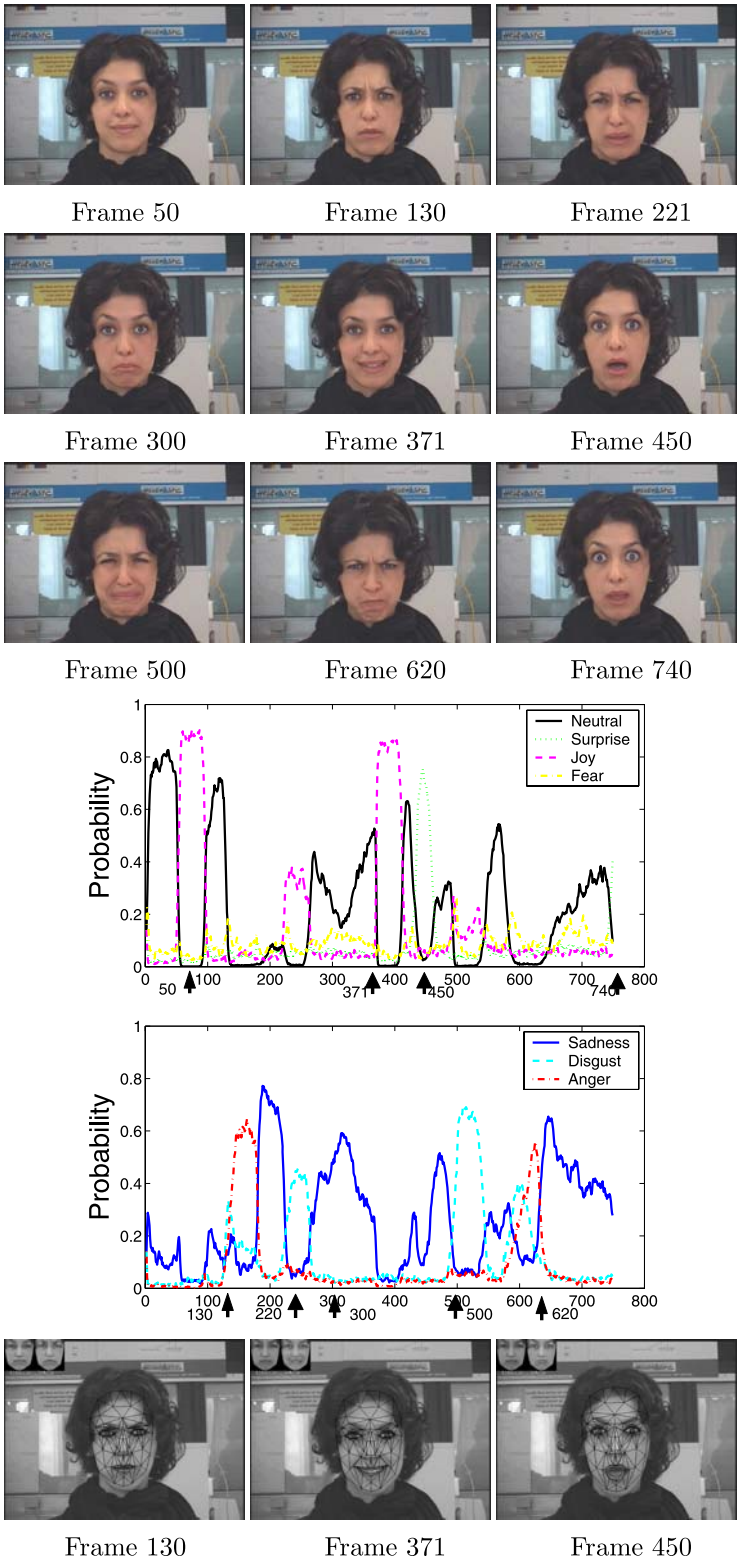


Fig. 8 Simultaneous tracking and recognition associated with a 748-frame video sequence. The top illustrates some frames of the test video. The *middle plots* shows the probability of each expression as a function of time (frames). The *bottom images* show the tracked facial actions where the corner shows the appearance mean and the current shape-free patch

ers may fail to keep track of them. In order to assess the behavior of our developed tracker whenever very rapid facial movements occur, we conducted the following experiment to simulate an ultra rapid mouth motion.[1] We cut about 40 frames from a test video. These frames (video seg-

[1]This experiment also simulates a discontinuity in video streaming.

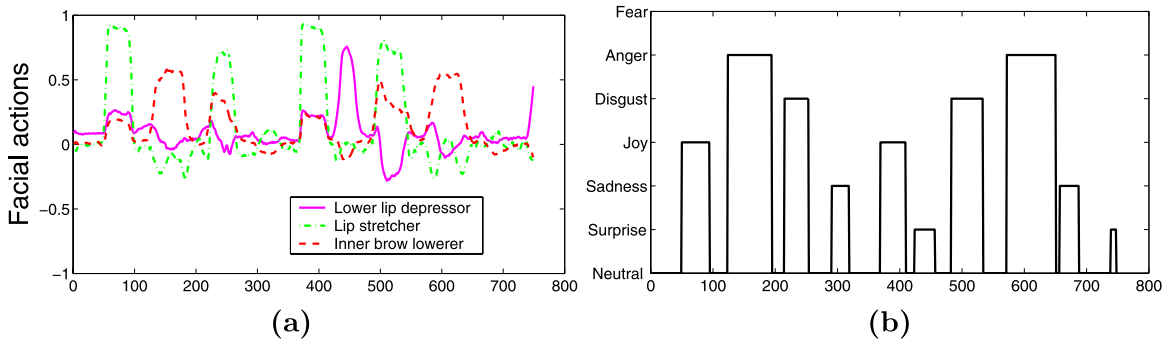


Fig. 9 **a** The tracked facial actions, $\hat{\boldsymbol{\tau}}_{\mathbf{a}(t)}$, computed by the recursive particle filter. **b** Segmenting the input video using the non-neutral frames

Fig. 10 The tracked facial actions, $\hat{\boldsymbol{\tau}}_{\mathbf{a}(t)}$ (weighted average), computed by the recursive particle filter with only 100 particles

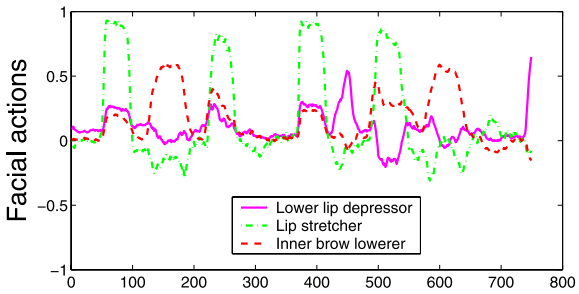


Fig. 11 Simultaneous tracking and recognition associated with a 600-frame video sequence depicting non-frontal head poses

ment) overlap with a surprise transition. The altered video is then tracked using two different methods: (i) a deterministic approach based on a registration technique estimating both the head and facial action parameters (Dornaika and Davoine 2006), and (ii) our stochastic approach. Figures 14(a) and 14(b) show the tracking results associated with the same input frame immediately after the cut. Note the difference in accuracy between the deterministic approach (a) and the stochastic one (b) (see the eyebrow and mouth region). Thus, despite the motion discontinuity of the mouth and the eyebrows, the particles are still able to provide the correct state (both the discrete and the continuous components) almost instantaneously (see the correct alignment between the 3D model and the region of the lips and mouth in Fig. 14(b)).

Low Resolution Video Sequences In order to assess the behavior of our developed approach when the resolution

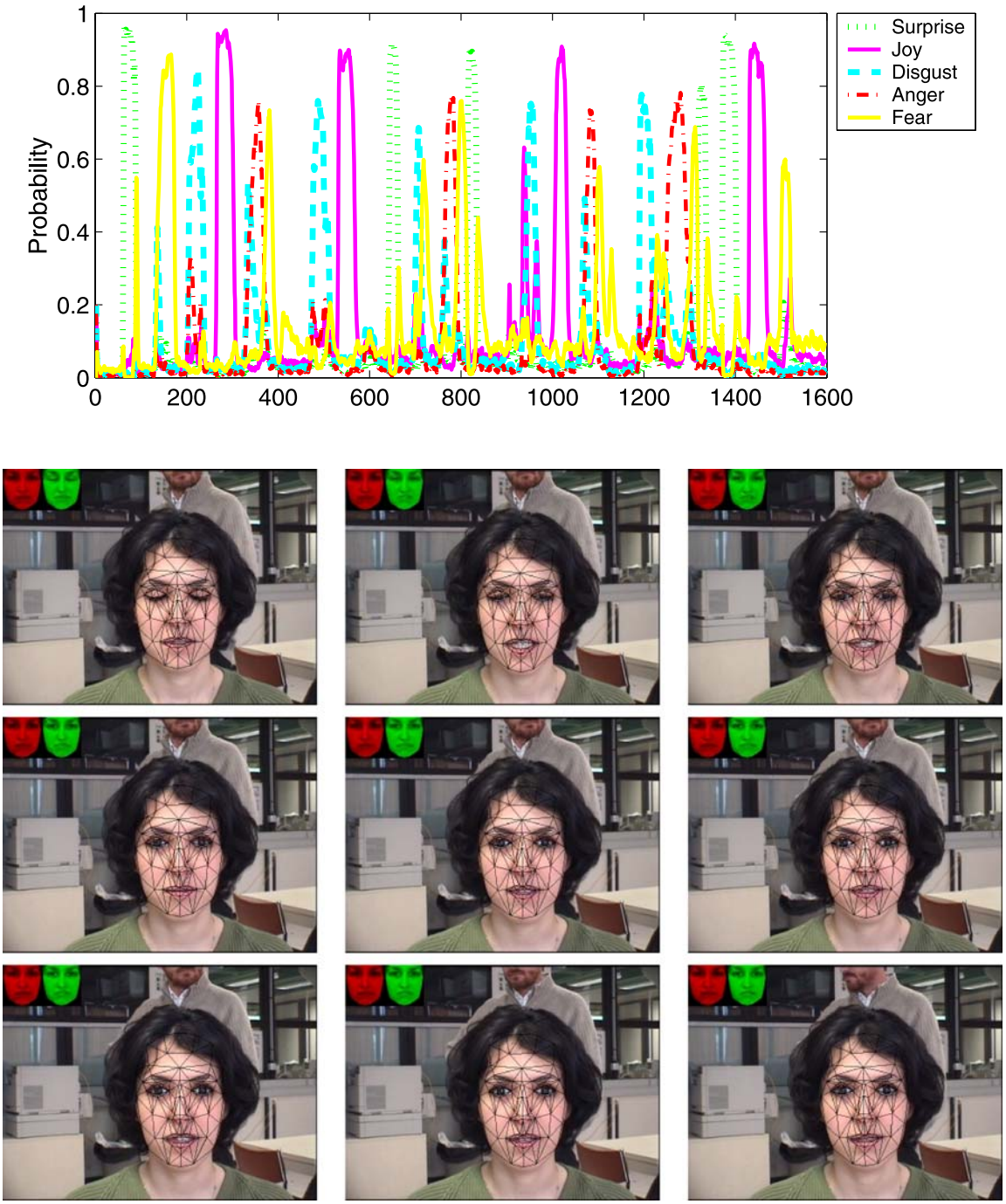


Fig. 12 *Top*: The probability of each expression as a function of time associated with a 1600-frame video sequence. *Bottom*: The tracked facial actions associated with the subject's speech which starts at frame 900 and ends at frame 930. Only frames 900, 903, 905, 907, 909, 911, 913, 917, and 925 are shown

and/or the quality of the videos is low, we downloaded several low-quality videos used in (Huang et al. 2002). In each 42-frame video, one universal expression is displayed. Figure 15 shows our recognition results (the discrete probability distribution) associated with three such videos. The left images display the 25^{th} frame of each video. Note that the neutral curve is not shown for reasons of clarity. As can be seen, the recognition obtained with our stochastic approach was very good despite the low quality of the videos used. The resolution of these videos is 320×240 pixels.

Moreover, we acquired several video sequences using the right camera of a low-cost stereo head—Point Grey Re-

Fig. 13 Method comparison: One class dynamics (**a**) versus multi-class dynamics (**b**) (see Sect. 5.2)

Fig. 14 Method comparison: Deterministic approach (**a**) versus our stochastic approach (**b**) immediately after a simulated mouth motion discontinuity (see Sect. 5.2)

search's Bumblebee sensor[2]. The captured 640×480 images are directly used by our tracker without any radial distortion correction. Figure 16 shows the tracking and recognition results associated with four frames. The actual expressions for these examples are indicated in bold type and the recognized ones (having the highest probability) are indicated in brackets. It will be remarked that these video sequences have combined three difficulties (1) the low quality of the images, (2) the non-frontal view of the face, and (3) the subject's facial dynamics were not learned beforehand. The first three images show three examples for which the facial expression recognition was correct (highest probability).

Although the subject's expression in the fourth image was one of disgust, the expression recognized was joy. This misrecognition is not related to the image quality nor to the non-frontal view but rather to the learned subject-specific dynamics. Indeed, whenever the tracker finds completely new dynamics it may fail to correctly recognize the source expression. Tackling this problem will be presented in Sect. 6.

Impact of Noisy Estimated 3D Head Pose The estimated appearance-based 3D head pose may suffer from some inaccuracies associated with the out-of-plane movements, which is the case for all monocular systems. It would seem reasonable to fear that these inaccuracies might lead to a failure in facial action tracking. In order to assess the effect of 3D head pose inaccuracies on the facial action tracking, we conducted the following experiment. We acquired a 750-frame sequence and performed our approach twice. The first was a straightforward run. In the second run, the estimated out-of-plane parameters of the 3D head pose were perturbed by a uniform noise, then the perturbed 3D pose was used by the facial action tracking and facial expression recognition. Figure 17 shows the value of the tracked actions in both cases: the noise-free 3D head pose (solid curve) and the noisy 3D head pose (dotted curves). In this experiment, the two out-of-plane angles were perturbed with additive uniform noise belonging to $[-7\text{degrees}, +7\text{degrees}]$ and the scale was perturbed by an additive noise belonging to $[-2\%, +2\%]$. As can be seen, the facial actions are almost not affected by the introduced noise. This can be explained by the fact that the 2D projection of out-of-plane errors produce very small errors in the image plane such that the 2D alignment between the model and the regions of lips and eyebrows is still good enough to capture their independent movements correctly.

[2]http://www.ptgrey.com.

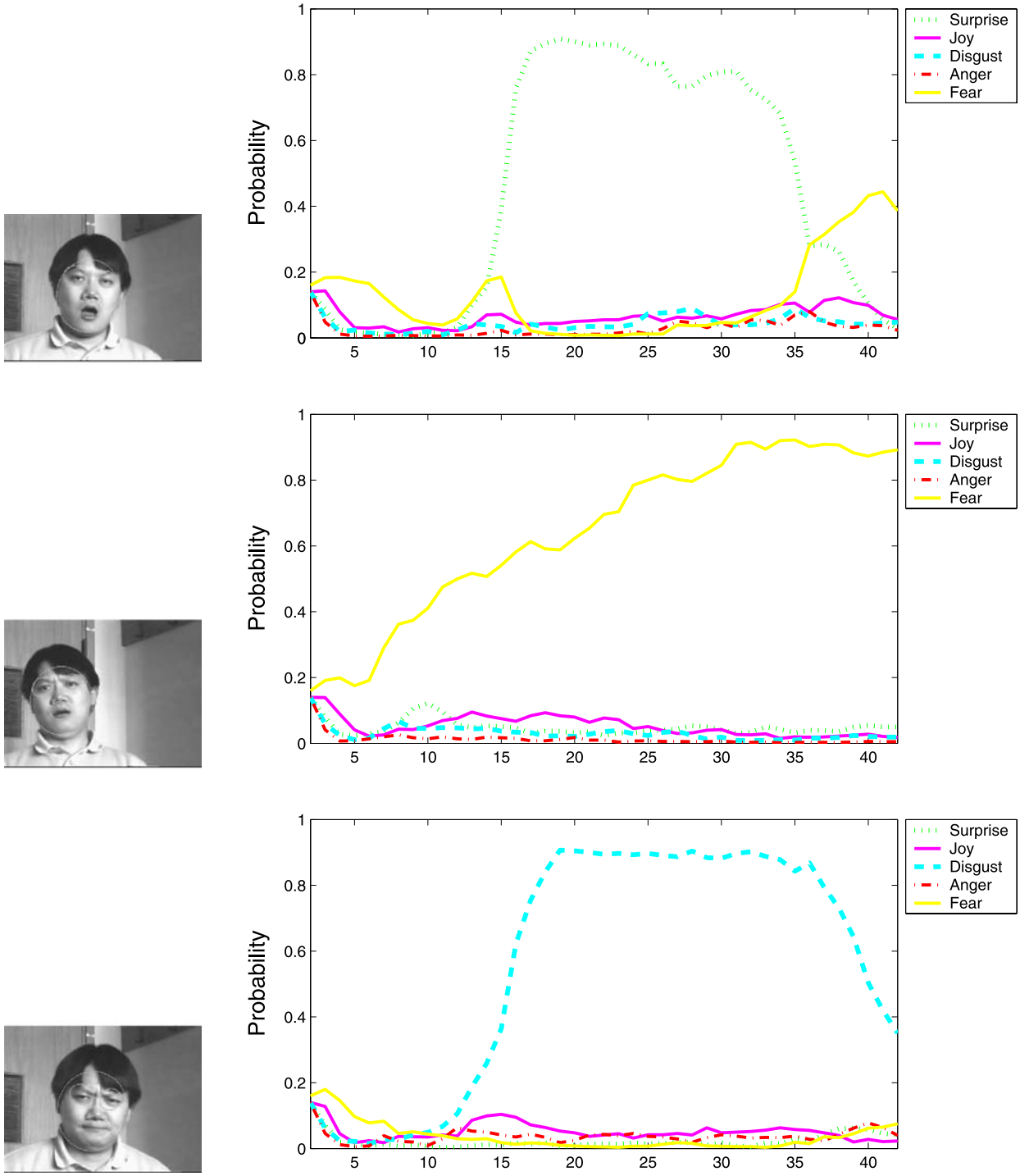


Fig. 15 The probability of each expression as a function of time associated with three low resolution videos. The *right images* display the 25^{th} frame of each video

Robustness to Lighting Conditions The appearance model used was given by one single multivariate Gaussian with parameters slowly updated over time. The robustness of this model is improved through the use of robust statistics that prevent outliers from deteriorating the global appearance model. This relatively simple model was adopted to allow real-time performance. We found that the tracking based on this model was successful even in the presence of temporary occlusions caused by a rotated face and occluding hands. Figure 18 illustrates the tracking results associated

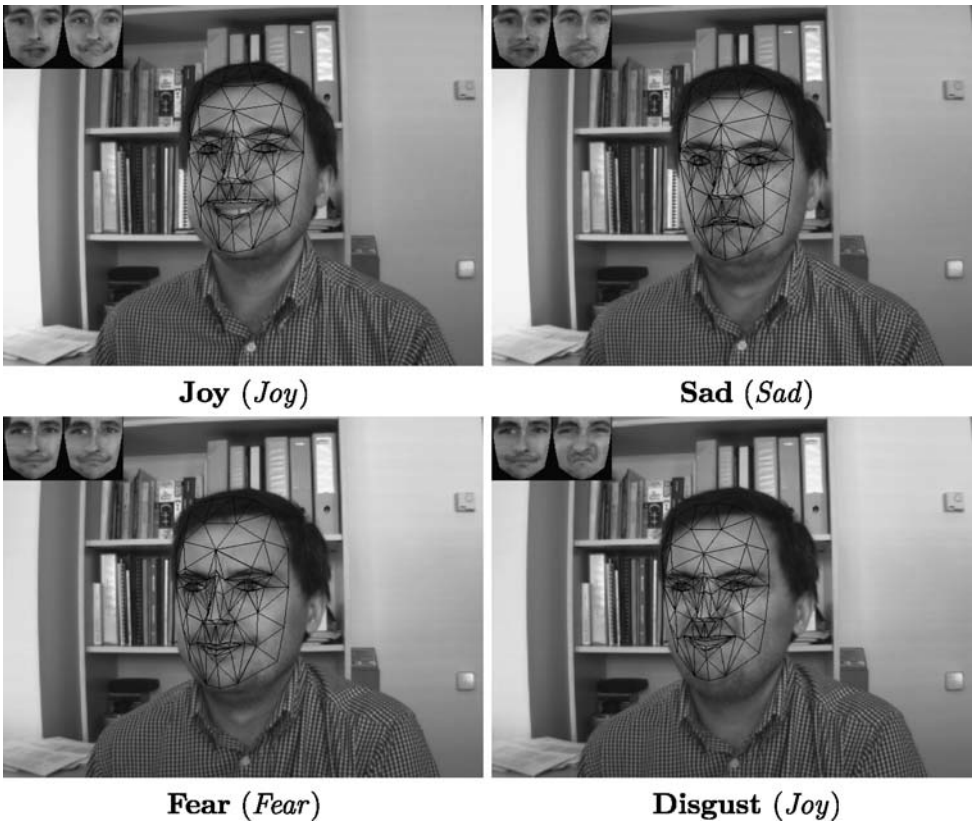


Fig. 16 Simultaneous tracking and recognition associated with low quality video sequences. The actual and recognized expressions are shown in bold and italics, respectively

with a video sequence provided by the Polytechnic University of Madrid,[3] depicting head movements and facial expressions under significant illumination changes. As can be seen, even though with our simple appearance model the possible brief perturbations caused temporary tracking inaccuracies, there is no track lost. Moreover, whenever the perturbation disappears the tracker begins once more to provide accurate parameters.

6 Subject-Dependent Dynamics

So far, the learned dynamic models were built using the facial actions associated with one human subject. However, it is known that the facial dynamics related to facial emotions and expressions are subject-dependent. For instance, expressions of anger and disgust may vary from one person to another. Therefore, the developed approach based on one person's data may fail to recognize the same expression in others.

In order to make the developed stochastic approach more general, we assume that the training data were produced by more than one person, each subject contributing a set of video sequences, where each video sequence depicts a given universal expression. Let N_{models} be the number of persons featured in the training data. The simplest way to merge data from several subjects is to concatenate the facial actions and to build one auto-regressive model per expression.

In our work, we will use another strategy. For each universal expression, we build N_{models} auto-regressive models (one for each subject). Thus each auto-regressive model will be given by $\mathbf{A}_1^{\gamma(n)}, \mathbf{A}_2^{\gamma(n)}$, $\mathbf{d}^{\gamma(n)}$, and $\mathbf{C}^{\gamma(n)} = \mathbf{B}^{\gamma(n)}\mathbf{B}^{\gamma(n)T}$ where γ denotes the expression and n denotes the person or model. Therefore, the total number of auto-regressive models is $(N_\gamma - 1)N_{\text{models}} + 1$, assuming that we adopt one AR model for the neutral expression.

The main steps of the simultaneous tracking and recognition method described in Sect. 4.2 remain the same. However, the prediction step will be totally different since each universal expression has several auto-regressive models. At run time, each particle label (expression) will be generated according to transition matrix probabilities, but the geometrical parameters (facial actions) will be generated by diffusion using all available N_{models} auto-regressive models associated with that expression label. However, only one predicted value for the facial actions will be retained: that which is most consistent with the observation likelihood. The developed approach will be broadly similar to the one presented in Fig. 7. However, for clarity of presentation we present it in Fig. 19.

[3] http://www.dia.fi.upm.es/~pcr/downloads.html.

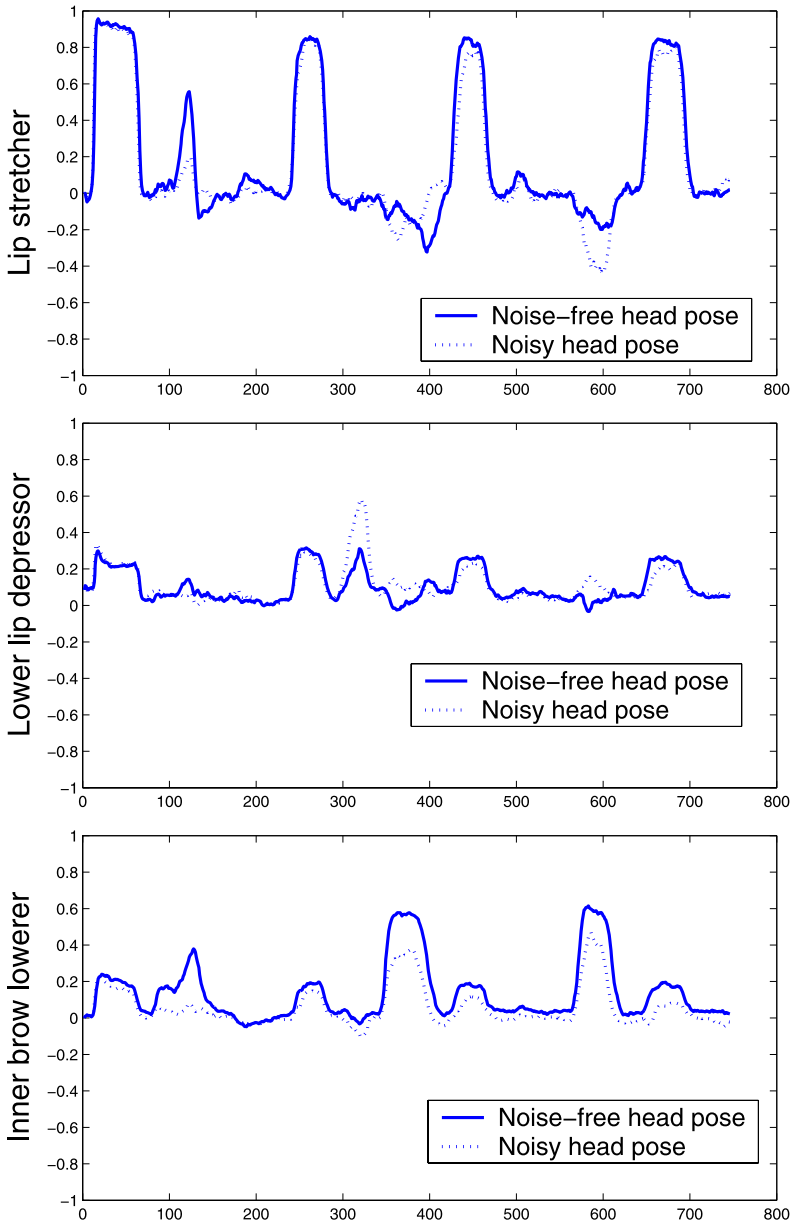


Fig. 17 Impact of noisy 3D head pose on the stochastic estimation of the facial actions. In each graph, the *solid curve* depicts the facial actions computed by the developed framework. The *dashed curve* depicts the same facial actions using a perturbed 3D pose

At first glance, one would expect CPU time to increase, since each predicted particle requires the computation of the observation likelihood (one computation per model). This time is considerably reduced by taking into account the fact that the generated particles differ only by the facial actions (they have the same 3D head pose parameters). So the observation likelihood (10) will be computed only for the pixels belonging to the triangles deformed by the facial actions and not for the whole facial patch.

We emphasize that reducing the computational load of the proposed algorithm (Fig. 19) when the number of persons is high can be done either by building one autoregressive model for all the subjects or by building a few autoregressive models that capture the main dynamics modes using unsupervised clustering techniques.

6.1 Experimental Results

We conducted several experiments to test the performance of the stochastic approach adopting several AR models per expression.

Figure 20 shows the tracking and recognition results associated with a 300-frame video sequence depicting some facial expressions displayed by an unseen person. Figure 20(a) shows the expression probability as a function of time using the stochastic approach with one AR model per expression (the training data correspond to one person).

Figure 20(b) displays the computed expression probability as a function of time when using $N_{\text{models}} = 2$ AR models per expression (the training data correspond to two persons). As can be seen (see frames 100 and 250), the anger

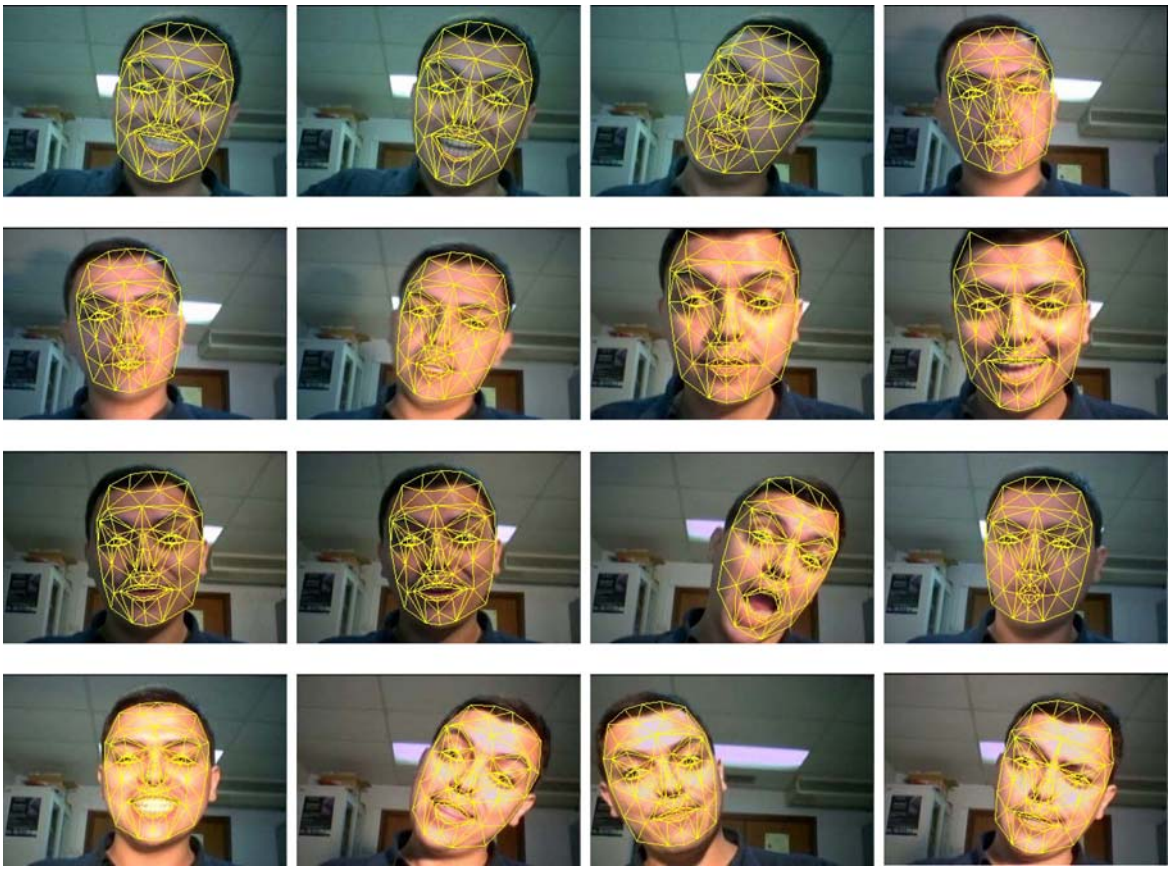


Fig. 18 Tracking the head and the facial actions under significant illumination changes and head and facial feature movements

expression is classified as a neutral one in (a) while the same expression is correctly recognized in (b). Figure 21 displays the computed facial actions associated with frame 250. (a) Displays the computed facial actions obtained with one AR model per expression. (b) Displays the computed facial actions obtained with two AR models per expression. As can be seen, the mouth is better tracked when two AR models per expression are used. Note that both recognition and tracking are improved by using several dynamics models.

Figure 22 shows the tracking and recognition results associated with a 300-frame video sequence depicting another unseen person. (a) Displays the probability of expression as a function of time using the stochastic approach with one AR model per expression. (b) Displays the probability of expression as a function of time when using two AR models per expression. As can be seen (for frames 75 through 125), disgust has been classified as joy in (a) while the same expression has been correctly recognized in (b).

In order to get quantitative evaluation of the proposed method, we proceeded as follows. We captured 25 video sequences depicting five universal expressions where each expression is displayed in five video sequences. All these expressions are performed by an unseen person. Table 1 displays the corresponding confusion matrix. Here an expression is correctly recognized if at least one of the two conditions is satisfied: (1) the transition neutral to that expression is correctly recognized, (2) the majority of the frames within duration of the expression have been correctly recognized. Recall that each frame is labeled with the expression having the maximum discrete probability.

Table 1 Confusion matrix obtained with the developed method. The learned AR models correspond to two subjects and the test data correspond to an unseen one

	Surprise	Joy	Disgust	Anger	Fear
Surprise	5	0	0	0	0
Joy	0	5	0	0	0
Disgust	0	0	4	0	0
Anger	0	0	0	5	0
Fear	0	0	1	0	5

1. **Initialization** $t = 0$:
   - Initialize the 3D head pose $\hat{\mathbf{h}}_0$
   - Generate J state samples $\mathbf{a}_0^{(1)}, \ldots, \mathbf{a}_0^{(J)}$ according to some prior density $p(\mathbf{a}_0)$ and assign them identical weights, $w_0^{(1)} = \ldots = w_0^{(J)} = 1/J$
2. **Tracking** At time step $t \leftarrow t+1$, get the input frame $\mathbf{y}_t$. Compute the corresponding 3D head pose, $\hat{\mathbf{h}}_t$, using the deterministic method outlined in Section 4.1. We have J weighted particles $(\mathbf{a}_{t-1}^{(j)}, w_{t-1}^{(j)})$ that approximate the posterior distribution of the state $p(\mathbf{a}_{t-1}|\mathbf{x}_{1:(t-1)})$ at the previous time step
   (a) Resample the particles proportionally to their weights, *i.e.* particles with high weights are duplicated and particles with small weights are removed. Resampled particles have the same weights
   (b) Draw J particles $\mathbf{a}_t^{(j)}$ according to the dynamic model $p(\mathbf{a}_t|\mathbf{a}_{t-1} = \mathbf{a}_{t-1}^{(j)})$. The obtained new particles approximate the predicted distribution $p(\mathbf{a}_t|\mathbf{x}_{1:(t-1)})$. For multi-class dynamics and mixed states this is done in two steps
   Discrete: Draw an expression label $\gamma_t^{(j)} = \gamma \in \mathcal{E} = \{1, 2, \ldots, N_\gamma\}$ with probability $T_{\gamma',\gamma}$, where $\gamma' = \gamma_{t-1}^{(j)}$
   Continuous: Compute $\boldsymbol{\tau}_{\mathbf{a}(t)}^{(j)}$ in two stages
   - For $n = 1, \ldots, N_{models}$ compute
     $$\boldsymbol{\tau}_{\mathbf{a}(t)}^{(n)} = \mathbf{A}_2^{\gamma(n)} \boldsymbol{\tau}_{\mathbf{a}(t-2)}^{(j)} + \mathbf{A}_1^{\gamma(n)} \boldsymbol{\tau}_{\mathbf{a}(t-1)}^{(j)} + \mathbf{d}^{\gamma(n)} + \mathbf{B}^{\gamma(n)} \mathbf{w}_t^{(j)}$$
     where $\gamma = \gamma_t^{(j)}$ and $\mathbf{w}_t^{(j)}$ is a 6-vector of standard normal variables
   - $\boldsymbol{\tau}_{\mathbf{a}(t)}^{(j)} = \boldsymbol{\tau}_{\mathbf{a}(t)}^{(n^\star)}$ where $n^\star = \arg\max_n \quad p(\mathbf{x}_t|\hat{\mathbf{h}}_t, \boldsymbol{\tau}_{\mathbf{a}(t)}^{(n)})$
   (c) Compute the shape-free patch $\mathbf{x}(\mathbf{b}_t^{(j)})$ according to (6) where $\mathbf{b}_t^{(j)} = [\ \hat{\mathbf{h}}_t^T, \quad \boldsymbol{\tau}_{\mathbf{a}(t)}^{(j)T}\ ]^T$
   (d) Weight each new particle proportionally to its likelihood
   $$w_t^{(j)} = \frac{p(\mathbf{x}_t|\mathbf{b}_t^{(j)})}{\sum_{m=1}^{J} p(\mathbf{x}_t|\mathbf{b}_t^{(m)})}$$
   The set of weighted particles approximates the posterior $p(\mathbf{a}_t|\mathbf{x}_{1:t})$
   (e) Set the probability of each basic expression $\gamma^\star \in \mathcal{E} = \{1, 2, \ldots, N_\gamma\}$ to
   $$P(\gamma^\star) = \sum_{m=1}^{J} \begin{cases} w_t^{(m)} & \text{if } \gamma_t^{(m)} = \gamma^\star \\ 0 & \text{otherwise} \end{cases}$$
   (f) Set the facial actions to $\hat{\boldsymbol{\tau}}_{\mathbf{a}(t)} = \sum_{m=1}^{J} w_t^{(m)} \boldsymbol{\tau}_{\mathbf{a}(t)}^{(m)}$
   (g) Set the geometrical parameters as $\hat{\mathbf{b}}_t = [\hat{\mathbf{h}}_t^T, \quad \hat{\boldsymbol{\tau}}_{\mathbf{a}(t)}^T\ ]^T$
   (h) Based on $\hat{\mathbf{b}}_t$, update the appearance (using Eqs. (9), (12), and (13)) as well as the 3D pose gradient matrix (Section 4.1). Go to 2.

Fig. 19 Inferring the 3D head pose, the facial actions and expression. A particle-filter-based algorithm is used for the simultaneous recovery of the facial actions and expression. The differences from the algorithm of Fig. 7 are shown in *black* and in a *large font*

7 Discussions

In this paper, we have proposed a stochastic framework that carries out simultaneously the tracking of facial actions and the recognition of expressions: a hard problem. Experiments show the robustness of the proposed method. The method is view-independent and does not require any learned facial image patch since the facial patch is learned online. The

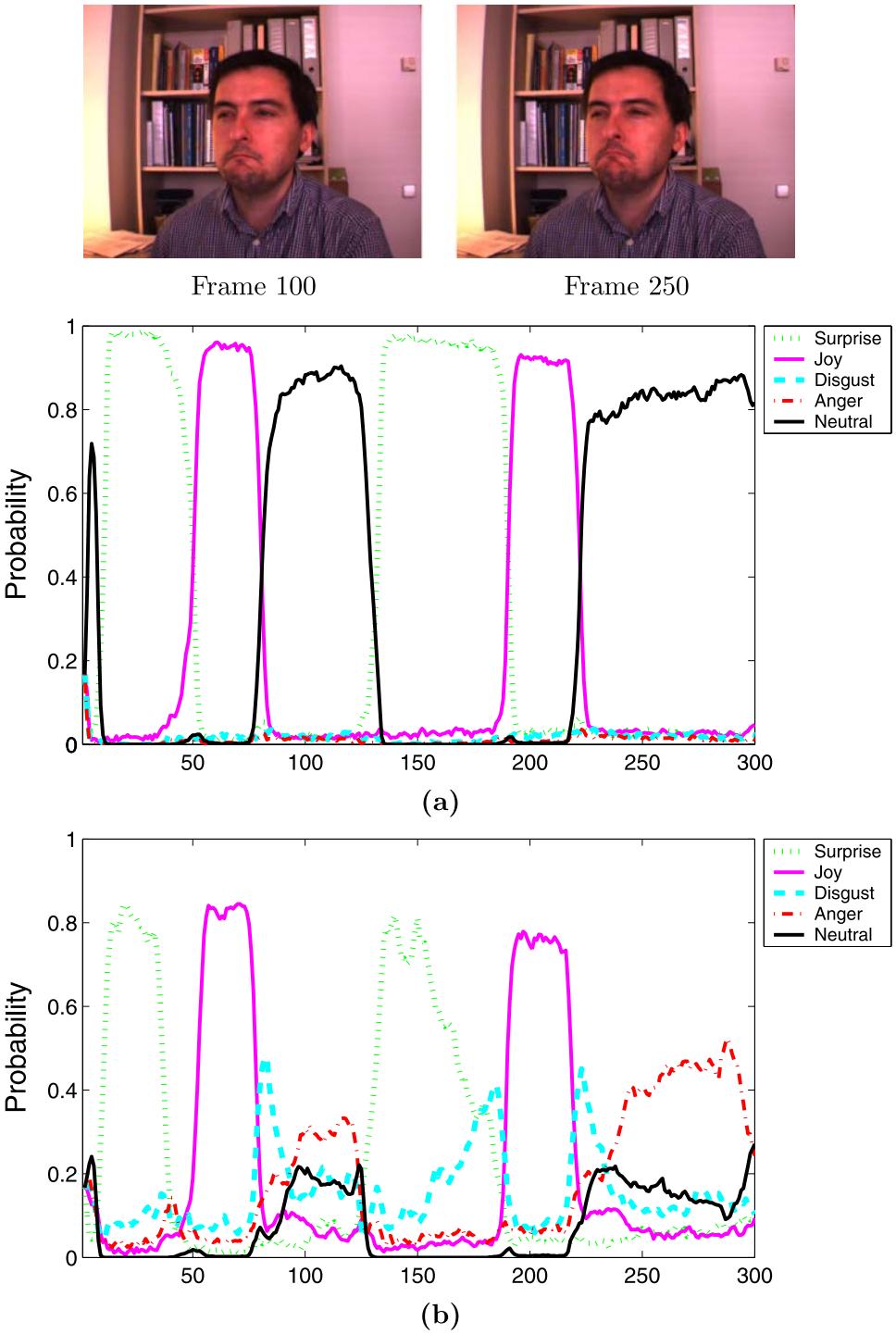


Fig. 20 A 300-frame video sequence depicting an unseen person's facial expressions. **a** The probability of expression as a function of time using the stochastic approach with one AR model per expression. **b** The probability of expression as a function of time when using two AR models per expression. As can be seen (see frames 100 and 250), the anger expression has been classified as a neutral one in (**a**) while the same expression has been correctly recognized in (**b**)

latter property makes it more flexible than many developed approaches. The proposed method can easily include other facial gestures in addition to the universal expressions. For each frame in the video sequence, our approach is split into two consecutive stages. In the first stage, the 3D head pose is recovered using a deterministic registration technique based on Online Appearance Models. In the second stage, the facial actions as well as the facial expression are simultaneously obtained using a stochastic framework based on multi-class dynamics.

We have shown that possible inaccuracies affecting the out-of-plane parameters associated with the 3D head pose have no impact on the stochastic tracking and recognition. The developed scheme lends itself nicely to real-time systems. Recall that the reported CPU time is associated with the deterministic method estimating the 3D head pose para-

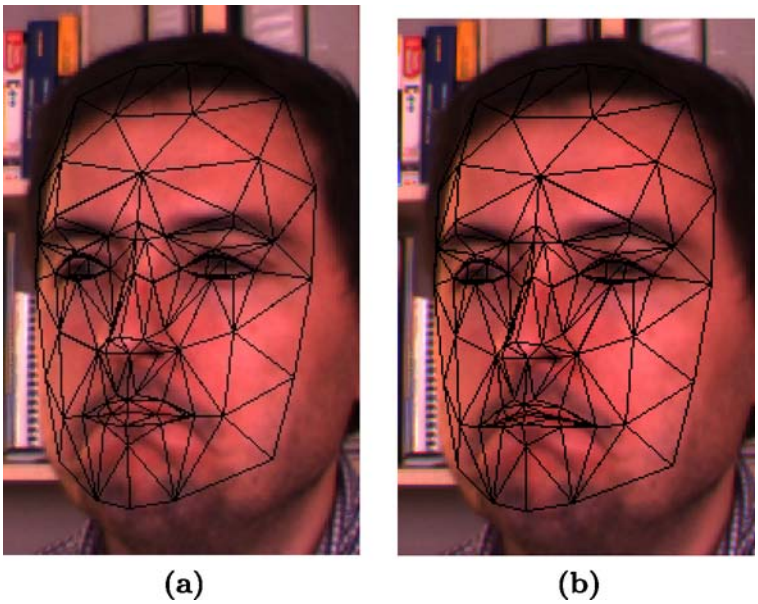


Fig. 21 The tracked facial actions associated with the frame 250 of the above sequence. **a** Depicts the tracking results obtained with one AR model per expression. **b** Depicts the tracking results obtained with two AR models per expression

meters and with the stochastic method estimating the facial actions and the expression.

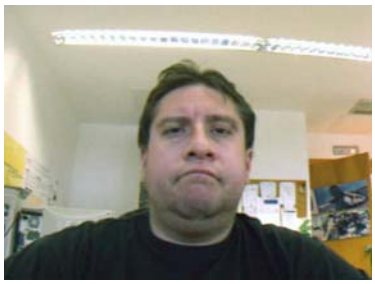
Moreover, we have proposed a scheme that is able to take into account the subject-dependent dynamics. This scheme makes the proposed framework more general. However, the strength of the stochastic approach is better exploited when the same person or dynamics are used. This fact is consistent with many researchers' findings that stipulate that temporal expression classifiers are very accurate when they deal with the same person.

We expect the approach to perform well in the presence of perturbing factors, such as video discontinuities and moderate illumination changes. Another advantage of our method is that it can track facial actions accurately even in the case where these actions do not belong to any learned facial gesture or expression. Certainly, in this case the corresponding non-learned gesture is classified as being the most similar learned one. Our tracker was successfully tested with moderate rapid head movements. Should ultra-rapid head movements break tracking, it is possible to use a re-initialization process or a stochastic tracker that propagates a probability distribution over time, such as the particle-filter-based tracking method presented in our previous work (Dornaika and Davoine 2006). The out-of-plane face motion range is limited within the interval [−45 deg, 45 deg] for the pitch and the yaw angles. Within this range, the obtained distortions associated with the facial patch are still acceptable to estimate the correct pose of the head.

The current work uses an appearance model given by one single multivariate Gaussian whose parameters are slowly updated over time. The robustness of this model is improved through the use of robust statistics that prevent outliers from deteriorating the global appearance model. This relatively simple model was adopted to allow real-time performance. We found that the tracking based on this model was successful even in the presence of occlusions caused by a rotated face and occluding hands. The current appearance model can be made more sophisticated through the use of Gaussian mixtures (Zhou et al. 2004; Lee 2005) and/or illumination templates to take into account sudden and significant local appearance changes due for instance to the presence of shadows.

In the current work, the head pose as well as the facial actions are manually initialized. In other words, the model parameters associated with the first frame in the video are manually set by an operator. Note that the proposed algorithm does not require that the first frame should be a neutral face since all universal expressions have the same probability. Future work will investigate building a full automatic tracker allowing the automatic initialization of the 3D head pose parameters and the facial actions. This can be done by using the principles of Active Appearance Models together with a global optimizer. Other automatic face and feature point detection algorithms may be used for initialization, using for example Haar feature-based AdaBoost classifiers combined with statistical shape models.

In our study, we tracked facial actions associated with the mouth and the eyebrows only. Many studies have shown that image regions associated with the mouth and the eyebrows are the most informative regions about the facial expressions. Certainly, the configuration of the eyelids is affected by the surprise and joy expressions. However, the movements of the mouth and the eyebrows are more informative than those associated with the eyelids.



Frame 110

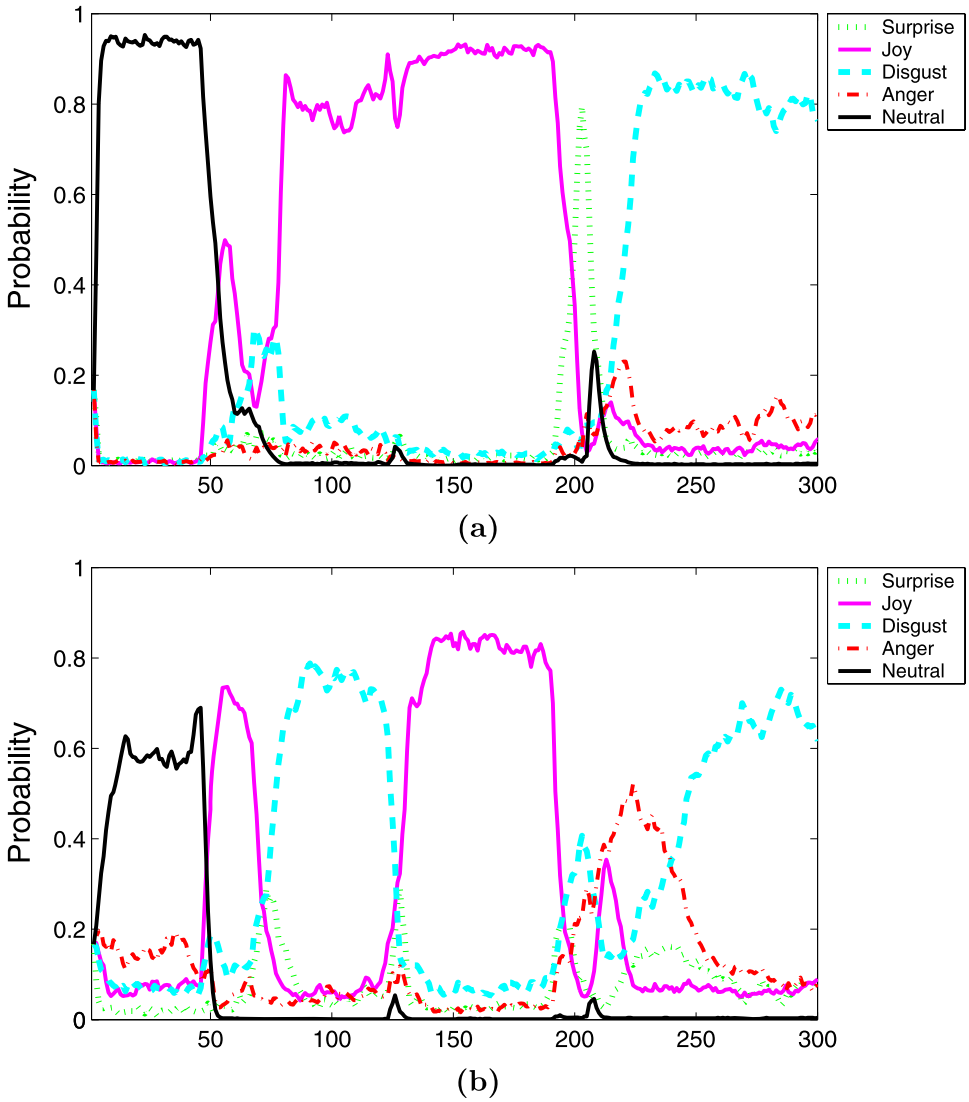


Fig. 22 A 300-frame video sequence depicting an unseen person's facial expressions. **a** The probability of expression as a function of time using the stochastic approach with one AR model per expression. **b** The probability of expression as a function of time when using two AR models per expression. As can be seen (for frames 75 through 125), the disgust expression has been classified as a joy one in (**a**) while the same expression has been correctly recognized in (**b**)

Moreover, we believe that iris movements will not provide significant information on the subject's facial expression.

It would be more elegant if one was able to incorporate the estimation of the 3D head pose parameters into the proposed particle filter. From a theoretical point of view this is indeed possible. However, in practice there are two major reasons that make this extremely challenging. First, head dynamics are quasi-totally independent from facial expression. For example, the head of someone showing anger expression can be still or undergoing arbitrary movements which might be very slow, very fast, or somewhere in between. Second, it is extremely challenging to compute a universal model for head motion dynamics.

Appendix

Given the time series $\boldsymbol{\tau}_{\mathbf{a}(t)}$ associated with a given expression γ, the AR model parameters $\mathbf{A}_1^\gamma, \mathbf{A}_2^\gamma, \mathbf{d}^\gamma, \mathbf{B}^\gamma$ are given by ($T' = T - 2$)

$$
\begin{aligned}
&(\mathbf{A}_1^\gamma \; \mathbf{A}_2^\gamma) = \overline{\mathbf{R}}_0 (\overline{\mathbf{R}})^{-1}, \\
&\mathbf{d}^\gamma = \frac{1}{T'}(\mathbf{R}_0 - \mathbf{A}\mathbf{R}), \\
&\mathbf{C}^\gamma = \frac{1}{T'}(\mathbf{R}_{0,0} - \mathbf{A}\overline{\mathbf{R}}_0^T)
\end{aligned}
\tag{17}
$$

where $\mathbf{C}^\gamma = \mathbf{B}^\gamma \mathbf{B}^{\gamma T}$ and

$$
\overline{\mathbf{R}} = \begin{pmatrix} \overline{\mathbf{R}}_{1,1} & \overline{\mathbf{R}}_{1,2} \\ \overline{\mathbf{R}}_{2,1} & \overline{\mathbf{R}}_{2,2} \end{pmatrix}; \qquad \overline{\mathbf{R}}_0 = (\, \overline{\mathbf{R}}_{0,1} \quad \overline{\mathbf{R}}_{0,2} \,);
$$

$$
\mathbf{R} = \begin{pmatrix} \mathbf{R}_0 \\ \mathbf{R}_1 \end{pmatrix}
$$

and the first-order moments $\mathbf{R}_i$ and autocorrelations $\mathbf{R}_{i,j}$ are given by (for clarity, we have omitted the superscript γ associated with these matrices)

$$
\begin{aligned}
&\mathbf{R}_i = \sum_{t=3}^{T} \boldsymbol{\tau}_{\mathbf{a}(t-i)}, \\
&\mathbf{R}_{i,j} = \sum_{t=3}^{T} \boldsymbol{\tau}_{\mathbf{a}(t-i)} \boldsymbol{\tau}_{\mathbf{a}(t-j)}^T, \\
&\overline{\mathbf{R}}_{i,j} = \mathbf{R}_{i,j} - \frac{1}{T'} \mathbf{R}_i \mathbf{R}_j^T.
\end{aligned}
$$

Note that the matrix $\mathbf{B}^\gamma$ can be obtained by a Cholesky factorization of the matrix $\mathbf{C}^\gamma$.

References

Ahlberg, J. (2002). An active model for facial feature tracking. *EURASIP Journal on Applied Signal Processing*, *2002*(6), 566–571.

Bartlett, M., Littlewort, G., Lainscsek, C., Fasel, I., & Movellan, J. (2004). Machine learning methods for fully automatic recognition of facial expressions and facial actions. In *IEEE international conference on systems, man and cybernetics*.

Bascle, B., & Black, A. (1998). Separability of pose and expression in facial tracking and animation. In *Proceedings of the IEEE international conference on computer vision*.

Blake, A., & Isard, M. (2000). *Active contours*. Berlin: Springer.

Blanz, V., & Vetter, T. (2003). Face recognition based on fitting a 3D morphable model. In *IEEE transactions on pattern analysis and machine intelligence* (pp. 1–12), September 2003.

Chandrasiri, N. P., Naemura, T., & Harashima, H. (2004). Interactive analysis and synthesis of facial expressions based on personal facial expression space. In *IEEE international conference on automatic face and gesture recognition*.

Cohen, I., Sebe, N., Garg, A., Chen, L., & Huang, T. S. (2003). Facial expression recognition from video sequences: Temporal and static modeling. *Computer Vision and Image Understanding*, *91*(1–2), 160–187.

Dornaika, F., & Davoine, F. (2005a). Simultaneous facial action tracking and expression recognition using a particle filter. In *IEEE international conference on computer vision*.

Dornaika, F., & Davoine, F. (2005b). View- and texture-independent facial expression recognition in videos using dynamic programming. In *IEEE international conference on image processing*.

Dornaika, F., & Davoine, F. (2006). On appearance based face and facial action tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(9), 2006.

Ekman, P., & Friesen, W. V. (1977). *Facial action coding system*. Palo Alto: Consulting Psychology Press.

Fasel, B., & Luettin, J. (2003). Automatic facial expression analysis: a survey. *Pattern Recognition*, *36*(1), 259–275.

Gokturk, S. B., Bouguet, J. Y., Tomasi, C., & Girod, B. (2002). Model-based face tracking for view-independent facial expression recognition. In *IEEE international conference on automatic face and gesture recognition*.

Huang, Y., Huang, T. S., & Niemann, H. (2002). A region-based method for model-free object tracking. In *16th international conference on pattern recognition*.

Huber, P. J. (1981). *Robust statistics*. New York: Wiley.

Isard, M., & Blake, A. (1998). A mixed-state condensation tracker with automatic model-switching. In *Proceedings of the IEEE international conference on computer vision*.

Jepson, A. D., Fleet, D. J., & El-Maraghi, T. F. (2003). Robust online appearance models for visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(10), 1296–1311.

Lee, D. (2005). Effective Gaussian mixture learning for video background subtraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(5), 827–832.

Liao, W.-K., & Cohen, I. (2005). Classifying facial gestures in presence of head motion. In *IEEE workshop on vision for human-computer interaction*.

Ljung, L. (1987). *System identification: theory for the user*. New York: Prentice Hall.

Lu, L., Zhang, Z., Shum, H. Y., Liu, Z., & Chen, H. (2001). Model- and exemplar-based robust head pose tracking under occlusion and varying expression. In *Proceedings of the IEEE workshop on models versus exemplars in computer vision (CVPR'01)*.

Lu, X., Jain, A. K., & Colbry, D. (2006). Matching 2.5D face scans to 3D models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(1), 31–43.

Lyons, M. J., Budynek, J., & Akamatsu, S. (1999). Automatic classification of single facial images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *21*(12), 1357–1362.

Moreno, F., Tarrida, A., Andrade-Cetto, J., & Sanfeliu, A. (2002). 3D real-time tracking fusing color histograms and stereovision. In *IEEE international conference on pattern recognition*.

North, B., Blake, A., Isard, M., & Rittscher, J. (2000). Learning and classification of complex dynamics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*(9), 1016–1034.

Pantic, M., & Rothkrantz, L. J. M. (2000). Automatic analysis of facial expressions: the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*(12), 1424–1445.

Perez, P., & Vermaak, J. (2005). Bayesian tracking with auxiliary discrete processes. Application to detection and tracking of objects with occlusions. In *IEEE ICCV workshop on dynamical vision*, Beijing, China.

Tian, Y., Kanade, T., & Cohn, J. F. (2001). Recognizing action units for facial expression analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*, 97–115.

Wang, Y., Ai, H., Wu, B., & Huang, C. (2004). Real time facial expression recognition with Adaboost. In *IEEE international conference on pattern recognition*.

Wen, Z., & Huang, T. S. (2003). Capturing subtle facial motions in 3D face tracking. In *IEEE international conference on computer vision*.

Yacoob, Y., & Davis, L. S. (1996). Recognizing human facial expressions from long image sequences using optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *18*(6), 636–642.

Yilmaz, A., Shafique, K. H., & Shah, M. (2002). Estimation of rigid and non-rigid facial motion using anatomical face model. In *IEEE international conference on pattern recognition*.

Zhang, Y., & Ji, Q. (2005). Active and dynamic information fusion for facial expression understanding from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(5), 699–714.

Zhou, S., Krueger, V., & Chellappa, R. (2003). Probabilistic recognition of human faces from video. *Computer Vision and Image Understanding*, *91*(1–2), 214–245.

Zhou, S., Chellappa, R., & Mogghaddam, B. (2004). Visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Transactions on Image Processing*, *13*(11), 1473–1490.