Machine Vision and Applications (2008) 19:427–441
DOI 10.1007/s00138-007-0091-1

SPECIAL ISSUE PAPER

Evaluation of an appearance-based 3D face tracker using dense 3D data

Fadi Dornaika · Angel D. Sappa

Received: 14 April 2006 / Revised: 8 March 2007 / Accepted: 9 May 2007 / Published online: 25 July 2007

Abstract The ability to detect and track human heads and faces in video sequences can be considered as the finest level of any video surveillance system. In this paper, we introduce a general framework for evaluating our recent appearance-based 3D face tracker using dense 3D data. This tracker combines online appearance models with an image registration technique and can run in real-time and is drift insensitive. More precisely, accuracy and usability of this developed tracker are assessed using stereo-based range facial data from which ground truth 3D motions are computed. This evaluation quantifies the monocular tracker accuracy, and identifies its working range in 3D space. Additionally, this evaluation gives some hints on how the tracker can be fully exploited.

1 Introduction

The ability to detect and track human motion in video sequences is a key requirement in a great number of applications such as video surveillance, human–computer interaction and gesture recognition. The finest level of tracking focuses on head motions and facial gestures/expressions [10].

Almost all video surveillance systems have addressed the estimation of body motions in 2D and 3D. The parametric representation of these captured motions can be used to infer the kind of the performed action such as walking and running [17]. However, information about the face and facial gestures are not exploited. In the context of video surveillance one can argue that the facial gestures/expressions once separated from the body motion (e.g., stabilizing the body motion) can be complementary and in some cases they can be more informative than the body motion. The main challenge of the classical video surveillance systems is the lack of resolution by which faces are viewed. However, nowadays due the advent of advanced sensors which may include active sensors or a network of cameras [6], the use of facial images in video surveillance systems is becoming feasible.

In our laboratory, we are building a three-level tracker. As illustrated in Fig. 1, the main procedure will be:

(a) detection and tracking of persons while they are still some distance away from the camera;
(b) when these persons come closer to the camera, or when the active camera zooms in on these persons, their body posture will be evaluated using state-of-the art human body motion trackers;
(c) if they are even closer and their face can be viewed with enough resolution, facial gestures and expressions will be tracked and inferred in order to see whether these are compatible with the assumptions made from their motions and postures.

1.1 Paper contribution

In this paper, we focus on the third level which concerns the tracking of the 3D face pose and some facial actions in monocular video sequences. More precisely, we will study the accuracy of a state-of-art monocular tracker. Vision-based 3D face tracking offers an attractive alternative since vision sensors are not invasive and hence natural motions can be achieved. However, detecting and tracking faces in video sequences is a challenging task because faces are non-rigid and their images have a high degree of variability. A huge

F. Dornaika (✉)
Institut Géographique National, 94165 Saint-Mandé, France
e-mail: fadi.dornaika@ign.fr

A. D. Sappa
Computer Vision Center, 08193 Bellaterra, Barcelona, Spain
e-mail: sappa@cvc.uab.es

Fig. 1 The three different types of human motion tracking that a global surveillance system should include

research effort has already been devoted to vision-based head and facial feature tracking in 2D and 3D (e.g., [7,12,14, 18,20,23]). Tracking the 3D head pose from a monocular image sequence is a difficult problem. Classical proposed techniques may be roughly classified into those based on optical flow and those based on tracking some salient features. Recently, researchers proposed deterministic and statistical appearance-based 3D head tracking methods which can successfully tackle the image variability and drift problems [1,7–9,19]. However, the accuracy of most of these developed approaches have not been quantitatively evaluated due to the lack of ground-truth data.

Recently, we have developed a fast and robust appearance-based 3D face tracker combining the concepts of online appearance models (OAMs) and image registration [9]. This tracker can provide the six degrees of freedom associated with the head pose as well as some facial actions. The proposed approach does not suffer from drifting and seems to be robust in the presence of large head motions and facial animations. In this paper, we summarize the developed approach and propose a general framework for evaluating the tracker accuracy based on dense depth data obtained from a stereo rig. The use of stereo rigs for inferring ground-truth 3D head motions has a big advantage over non-visual sensors and most of classical range sensors in the sense that these ones require a tedious calibration task in order to relate the non-visual sensor frame to the camera frame. The main innovation of this paper is the introduction of an evaluation of the appearance-based tracker using stereo-based dense range data. The tests are carried out on real video sequences provided by a stereo rig that (1) provides the tracker with the monocular sequences and (2) also provides dense range data used for recovering the ground truth 3D head motions.

The evaluation of our proposed appearance-based tracker has not been more formal than observing that it works quite well and that the features of the 3D model projects onto their corresponding 2D features in the image sequence. The problem with an objective evaluation is that the absolute truth is not known. This is particularly true for the 3D head pose/motion which is given by six degrees of freedom. However, it is less problematic for the facial feature motion since their estimated motion can be assessed by checking the alignment between the projected 3D model (feature points and line segments) and the actual location of the facial features. In our case, the facial features are given by the lips and the eyebrows so evaluating their motion is straightforward. We point out that their corresponding motions essentially belong to the frontal plane of the face. Moreover, since these features have different textures, their independent motion can be accurately recovered by the appearance-based tracker.

There are other techniques for measuring face motion, such as motion capture systems based on acoustic trackers [21] or magnetic sensors. However, such systems are expensive and encumbering, and may not succeed to capture small motion accurately. Since we are using a deformable 3D mesh we can adopt an inexpensive solution that employs synthetic test sequences with known ground truth similarly to [2,13]. In [2], a 3D face tracker based on a statistical facial texture was evaluated. A synthetic video sequence is created using a 3D mesh mapping a texture onto it, and then animating it according to some captured or semi-random motion. The tracker then tracks the face in the synthetic video sequence and the discrepancy between the used synthetic motion (ground-truth) and the estimated motion yields the accuracy of the tracker.

Although this scheme can give an idea on the tracker accuracy, it has several shortcomings. First, one can note the self-referential nature of the test, since the same 3D mesh is used in the synthesis phase and in the test phase. Second, synthetic videos may not look very life-like. Third, since the synthetic motion should be realistic to some extent, one has to use the output of another tracker, and if the same tracker is used the evaluation test becomes self-referential regarding the used 3D motions in the sense that the tracker is tested with motion parameters that are easy to estimate. Therefore, our idea is to use stereo-based 3D facial surfaces (from which an accurate rigid 3D head motion can be retrieved), and at the same time run our appearance based on the associated monocular sequence. Then, the accuracy is evaluated by comparing the 3D head motions provided by the developed monocular 3D face tracker and the ground-truth 3D head motions provided by stereo data. Since the 3D data associated with the face surface are accurate and since the used registration—the iterative closest point (ICP) algorithm—is performing a fine registration, the resulting 3D head motions can be considered as "ground-truth" data.

Notice that 3D face models can be obtained using active sensors such as [4,5,11,22]. The use of stereo rigs for inferring ground-truth 3D head motions has a big advantage over most of active range sensors in the sense that the latter ones require a tedious calibration task in order to relate the sensor frame to the camera frame since one has to express all 3D motions in a common coordinate system.

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, and the online reconstruction of the facial appearance model. In order to make the paper self-contained, Sect. 4 summarizes the principles of our recent appearance-based 3D face tracker, that is, the recovery of the 3D head pose and facial actions. Section 5 introduces the proposed accuracy evaluation framework based on 3D facial surfaces and gives quantitative accuracy evaluation obtained with real video sequences. Section 6 concludes the paper.

2 Modeling faces

2.1 A deformable 3D model

In our study, we use the 3D face model *Candide*. This 3D deformable wireframe model was first developed for the purpose of model-based image coding and computer animation. The 3D shape of this wireframe model 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 $\mathbf{g}$; the concatenation of the 3D coordinates of all vertices $\mathbf{P}_i$. The vector $\mathbf{g}$ is written as:

$$\mathbf{g} = \bar{\mathbf{g}} + \mathbf{S}\,\boldsymbol{\tau}_s + \mathbf{A}\,\boldsymbol{\tau}_a \tag{1}$$

where $\bar{\mathbf{g}}$ is the standard shape of the model, $\boldsymbol{\tau}_s$ and $\boldsymbol{\tau}_a$ are shape and animation control vectors, respectively, and the columns of $\mathbf{S}$ and $\mathbf{A}$ are the shape and animation units. A shape unit provides a way to deform the 3D wireframe such as to adapt the eye width, the head width, the eye separation distance, etc. Thus, the term $\mathbf{S}\,\boldsymbol{\tau}_s$ accounts for shape variability (inter-person variability) while the term $\mathbf{A}\,\boldsymbol{\tau}_a$ 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 $\boldsymbol{\tau}_a = \mathbf{0}$. We point out that since the evaluation process is based on the ICP registration technique, the videos used by the evaluation have to depict face motion without facial expressions. However, the monocular tracker estimates the 3D head pose parameters and the facial actions.

In this study, we use 12 modes for the facial shape units matrix $\mathbf{S}$ and six modes for the facial animation units animation units (AUs) matrix $\mathbf{A}$. 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.

In Eq. (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, $\mathbf{M}$, encapsulating both the 3D head pose and the camera parameters.

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

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

For a given person, $\boldsymbol{\tau}_s$ is constant. Estimating $\boldsymbol{\tau}_s$ can be carried out using either feature-based or featureless approaches. Thus, the state of the 3D wireframe model is given by the 3D head pose parameters (three rotations and three translations) and the internal face animation control vector $\boldsymbol{\tau}_a$. This is given by the 12-dimensional vector $\mathbf{b}$:

$$\mathbf{b} = [\theta_x, \theta_y, \theta_z, t_x, t_y, t_z, \boldsymbol{\tau}_a{}^{\mathrm{T}}]^{\mathrm{T}} \tag{3}$$

2.2 Shape-free facial patches

A face texture is represented as a shape-free texture (geometrically normalized image). The geometry of this image is obtained by projecting the standard shape $\bar{\mathbf{g}}$ using a centered frontal 3D pose onto an image with a given resolution. The texture of this 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, $\mathcal{W}$.

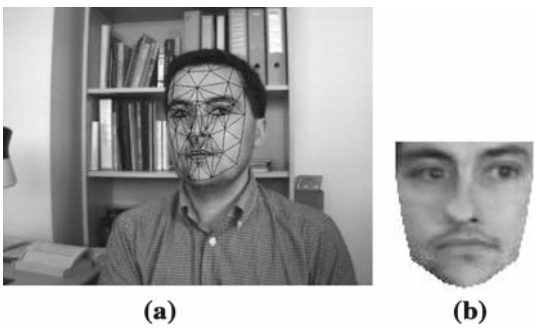

Fig. 2 **a** An input image with correct adaptation. **b** The corresponding shape-free facial image

The warping process applied to an input image $\mathbf{y}$ is denoted by:

$$\mathbf{x}(\mathbf{b}) = \mathcal{W}(\mathbf{y}, \mathbf{b}) \tag{4}$$

where $\mathbf{x}$ denotes the shape-free texture patch and $\mathbf{b}$ denotes the geometrical parameters. Several resolution levels can be chosen for the shape-free textures. The reported results are obtained with a shape-free patch of 5,392 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: (1) transfer the texture $\mathbf{y}$ using the piece-wise affine transform associated with the vector $\mathbf{b}$, and (2) perform the grey-level normalization of the obtained patch.

3 Problem formulation and adaptive observation model

Given a video sequence depicting a moving head/face, we would like to recover, for each frame, the 3D head pose and the facial actions encoded by the control vector $\boldsymbol{\tau_a}$. In other words, we would like to estimate the vector $\mathbf{b}_t$ [Eq. (3)] at time t given all the observed data until 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 handed over to the next frame.

For each input frame $\mathbf{y}_t$, the observation is simply the warped texture patch (the shape-free patch) associated with the geometric parameters $\mathbf{b}_t$. We use the HAT symbol for the tracked parameters and textures. 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,

$$\hat{\mathbf{x}}_t = \mathbf{x}(\hat{\mathbf{b}}_t) = \mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_t) \tag{5}$$

The estimation of $\hat{\mathbf{b}}_t$ from the sequence of images will be presented in the next section.

The appearance model associated with the shape-free facial patch at time t, is time varying in that it models the appearances present in all observations $\hat{\mathbf{x}}$ up to time $(t-1)$. This can be required due, for instance, to illumination changes and 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 $\boldsymbol{\Sigma}$. The choice of a Gaussian distribution is motivated by the fact that this kind of distribution provides 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 $\boldsymbol{\Sigma}$. $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are d-vectors (d is the size of $\mathbf{x}$). Although the independence assumption may be violated, at least locally, we adopt it in our work in order to keep the problem tractable.

In summary, the observation likelihood is written as

$$p(\mathbf{y}_t|\mathbf{b}_t) = p(\mathbf{x}_t|\mathbf{b}_t) = \prod_{i=1}^{d} \mathbf{N}(x_i; \mu_i, \sigma_i)_t \tag{6}$$

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

$$\mathbf{N}(x_i; \mu_i, \sigma_i) = (2\pi\sigma_i^2)^{-1/2} \exp\left[-\rho\left(\frac{x_i - \mu_i}{\sigma_i}\right)\right], \quad \rho(x) = \frac{1}{2}x^2 \tag{7}$$

We assume that the appearance model summarizes the past observations under an exponential envelope, that is, the past observations are exponentially forgotten with respect to the current texture. When the appearance is tracked for the current input image, i.e., the texture $\hat{\mathbf{x}}_t$ is available, we can compute the updated appearance and use it to track in the next frame.

It can be shown that the appearance model parameters, i.e., the μ_i and σ_i values can be updated from time t to time $(t+1)$ using the following equations (see [15] for more details on OAMs):

$$\mu_{i_{(t+1)}} = (1-\alpha)\,\mu_{i_{(t)}} + \alpha\,\hat{x}_{i_{(t)}} \tag{8}$$

$$\sigma^2_{i_{(t+1)}} = (1-\alpha)\,\sigma^2_{i_{(t)}} + \alpha\,(\hat{x}_{i_{(t)}} - \mu_{i_{(t)}})^2 \tag{9}$$

This technique, also called recursive filtering, 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. In this figure, the vertical coordinate denotes the blending weight associated with all previous frames. For example, the contribution of frame 450 to the cur-

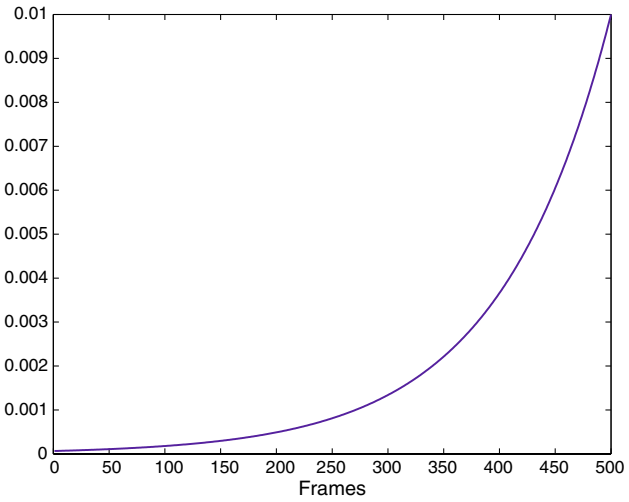

Fig. 3 A sliding exponential envelope having α equal to 0.01. The current frame/time is 500. The vertical coordinate corresponds to the blending weight

rent texture model is weighted by 0.006. Recall that the recursive Eqs. (8) and (9) are performing this blend implicitly without doing an explicit summation over the whole frames.

Note that $\boldsymbol{\mu}$ is initialized with the first patch $\mathbf{x}$. However, Eq. (9) is not used 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, Eq. (9) is used with α being set to $\frac{1}{t}$.

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 (e.g., see [16,24]).

4 Tracking with a registration technique

Consider the state vector $\mathbf{b} = [\theta_x, \theta_y, \theta_z, t_x, t_y, t_z, \boldsymbol{\tau_a}^{\mathrm{T}}]^{\mathrm{T}}$ encapsulating the 3D head pose and the facial animations. In this section, we will show how this state can be recovered for time t from the previous known state $\hat{\mathbf{b}}_{t-1}$.

The sought geometrical parameters $\mathbf{b}_t$ at time t are related to the previous parameters by the following equation ($\hat{\mathbf{b}}_{t-1}$ is known):

$$\mathbf{b}_t = \hat{\mathbf{b}}_{t-1} + \Delta\mathbf{b}_t \tag{10}$$

where $\Delta\mathbf{b}_t$ is the unknown shift in the geometric parameters. This shift is estimated using a region-based registration technique that does not need any image feature extraction. In other words, $\Delta\mathbf{b}_t$ is estimated such that the warped texture will be as close as possible to the facial appearance model. For this purpose, we minimize the *Mahalanobis* distance between the warped texture and the current appearance mean,

$$\min_{\mathbf{b}_t} e(\mathbf{b}_t) = \min_{\mathbf{b}_t} D(\mathbf{x}(\mathbf{b}_t),\ \boldsymbol{\mu}_t) = \sum_{i=1}^{d} \left(\frac{x_i - \mu_i}{\sigma_i}\right)^2 \tag{11}$$

The above criterion can be minimized using iterative first-order linear approximation which is equivalent to a Gauss–Newton method. It is worthwhile noting that minimizing the above criterion is equivalent to maximizing the likelihood measure given by Eq. (6).

4.1 Registration

We assume that there exists $\mathbf{b}_t = \hat{\mathbf{b}}_{t-1} + \Delta\mathbf{b}_t$ such that the warped shape-free texture will be very close to the appearance mean, i.e,

$$\mathcal{W}(\mathbf{y}_t, \mathbf{b}_t) \simeq \boldsymbol{\mu}_t$$

Approximating $\mathcal{W}(\mathbf{y}_t, \mathbf{b}_t)$ via a first-order Taylor series expansion around $\hat{\mathbf{b}}_{t-1}$ yields

$$\mathcal{W}(\mathbf{y}_t, \mathbf{b}_t) \simeq \mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_{t-1}) + \mathbf{G}_t(\mathbf{b}_t - \hat{\mathbf{b}}_{t-1})$$

where $\mathbf{G}_t$ is the gradient matrix. By combining the previous two equations we have:

$$\boldsymbol{\mu}_t = \mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_{t-1}) + \mathbf{G}_t\,(\mathbf{b}_t - \hat{\mathbf{b}}_{t-1})$$

Therefore, the shift in the parameter space is given by:

$$\Delta\mathbf{b}_t = \mathbf{b}_t - \hat{\mathbf{b}}_{t-1} = -\mathbf{G}_t^{\dagger}\,(\mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_{t-1}) - \boldsymbol{\mu}_t) \tag{12}$$

In practice, the solution $\mathbf{b}_t$ (or equivalently the shift $\Delta\mathbf{b}_t$) is estimated by running several iterations until the error cannot be improved. We proceed as follows.

Starting from $\mathbf{b} = \hat{\mathbf{b}}_{t-1}$, we compute the error vector $(\mathcal{W}(\mathbf{y}_t, \hat{\mathbf{b}}_{t-1}) - \boldsymbol{\mu}_t)$ and the corresponding *Mahalanobis* distance $e(\mathbf{b})$ [given by Eq. (11)]. We find a shift $\Delta\mathbf{b}$ by multiplying the error vector with the negative pseudo-inverse of the gradient matrix using Eq. (12). The vector $\Delta\mathbf{b}$ gives a displacement in the search space for which the error, e, can be minimized. We compute a new parameter vector and a new error:

$$\begin{aligned} \mathbf{b}' &= \mathbf{b} + \rho\,\Delta\mathbf{b} \\ e' &= e(\mathbf{b}') \end{aligned} \tag{13}$$

where ρ is a positive real.

If $e' < e$, we update $\mathbf{b}$ according to Eq. (13) and the process is iterated until convergence. If $e' \geq e$, we try smaller update steps in the same direction (i.e., a smaller ρ is used). Convergence is declared when the error cannot be improved anymore. In practice, we found that convergence is reached with less than ten iterations. The gradient matrix is built online using numerical differences.

4.2 Improving the minimized criterion

When significant out-of-plane rotations of the face occur, local self occlusions and distortions may appear in the shape-free texture $\mathbf{x}$. In order to downweight their influence on the registration technique we incorporate the orientation of individual triangles of the 3D mesh in the minimized criterion such that the contribution of any triangle becomes less significant as it shies away from the frontal view. Recall that the orientation of any 3D triangle with respect to the camera can be recovered since the 3D rotation between the 3D head model and the camera frame is tracked. For a given triangle, m, the angle γ_m is given by the angle between the optical axis $\mathbf{k} = (0, 0, 1)^{\mathrm{T}}$ and the normal to the triangle expressed in the camera frame.

For any given frame, the minimized criterion (11) becomes:

$$\min_{\mathbf{b}_t} e(\mathbf{b}_t) = \sum_{i=1}^{d} w(\gamma_i) \left(\frac{x_i - \mu_i}{\sigma_i}\right)^2 \tag{14}$$

where γ_i is the angle associated to the triangle containing the pixel i and $w(\gamma_i)$ is a monotonic decreasing function.

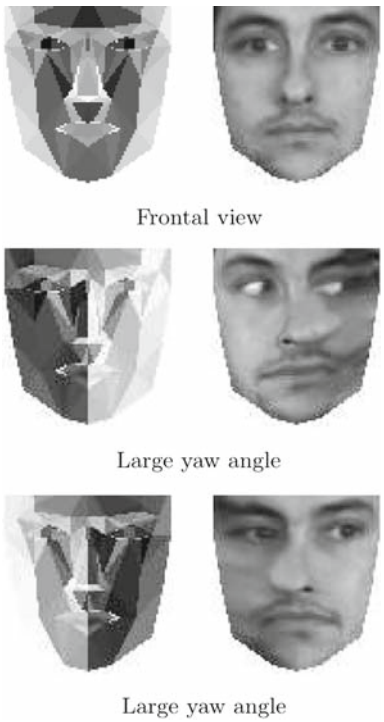

Fig. 4 Three different 3D face orientations (from top to bottom): frontal view, a vertical rotation to the left, and a vertical rotation to the right. The *first column* depicts the angle of all 3D triangles with respect to the camera. *Dark grey* level corresponds to small angles (the 3D triangle is in fronto-parallel plane) while *bright grey* level corresponds to large angles. The *right column* depicts the corresponding shape-free texture

For example, we use

$$w(\gamma_i) = \frac{1}{1+\gamma_j}$$

Figure 4 displays three real 3D face poses: a frontal view and two non-frontal views. The left column shows the orientation of all individual triangles of the 3D mesh. The right column shows the corresponding shape-free texture.

4.3 Tracking results

Figure 5 displays the head and facial action tracking results associated with a 300-frame video sequence (only two frames are shown). This video sequence depicts head motions and facial expressions. The upper left corner of each image shows the current appearance ($\boldsymbol{\mu}_t$) and the current shape-free texture ($\hat{\mathbf{x}}_t$).

On a 3.2 GHz PC, a non-optimized C code of the approach computes the 12 degrees of freedom (the six 3D head pose parameters and the six facial actions) in less than 50 ms if the patch resolution is 1,310 pixels. About half that time is required to compute the 3D head pose parameters.

Figures 6, 7, and 8 illustrate the estimated 3D head pose parameters associated with three 300-frame sequences. The three sequences are of resolution 640 × 480 pixels. Each figure illustrates a few frames of the video as well as the estimated 3D head pose parameters (the three rotations and the three translations) as a function of the sequence frames. These parameters are presented in six graphs. Since the used camera is calibrated the absolute translation is recovered. These video sequences will be used for accuracy evaluation of the monocular tracker using the framework proposed in Sect. 5.

5 Accuracy evaluation

5.1 3D facial data and ground-truth 3D face motions

A commercial stereo vision camera system (Bumblebee from Point Grey (http://www.ptgrey.com)) was used. It consists of two Sony ICX084 color CCDs with 6 mm focal length lenses. Bumblebee is a precalibrated system that does not require in-field calibration. The baseline of the stereo head is 12 cm and is connected to the computer by a IEEE-1394 connector. Right and left color images were captured at a resolution of 640×480 pixels and a frame rate near to 30 fps. After capturing these right and left images, 3D data were computed using the provided 3D reconstruction software. In our evaluation tests, the stereo rig was placed at a distance of 60–80 cm from the subject's head. Figure 9a shows a stereo pair used in our evaluation. Figure 9b shows the corresponding 3D facial data visualized from three different points of view. Figure 9c depicts the 3D data associated with another stereo pair depicting a non-frontal face. In our

Fig. 5 Tracking the 3D head pose and the facial actions. The upper left corner of each image shows the current appearance $\boldsymbol{\mu}_t$ and the current texture $\hat{\mathbf{x}}_t$

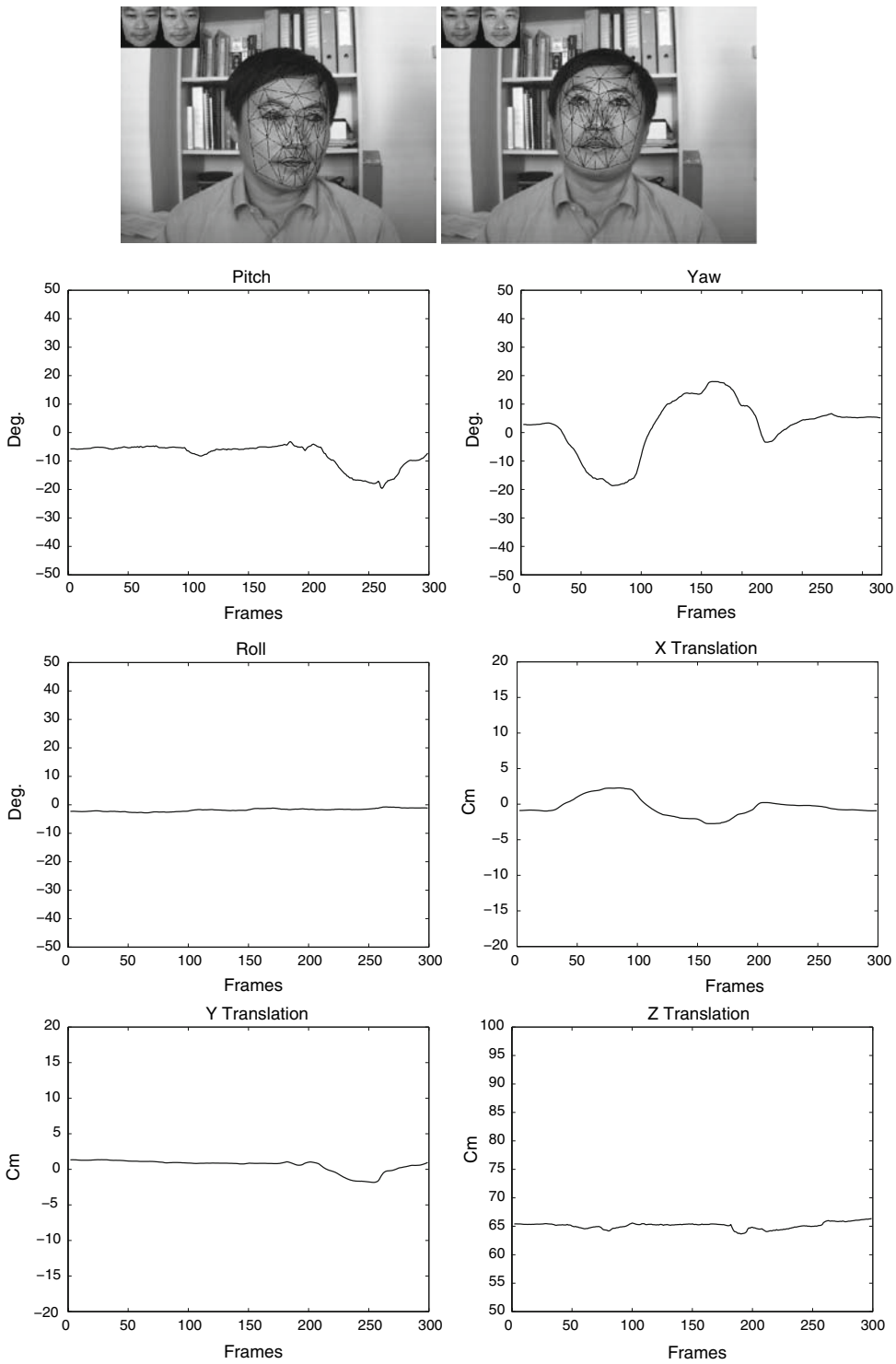

Fig. 6 Tracking the 3D head pose parameters associated with the first video sequence. Only frames 81 and 244 are shown. The six plots display the estimated six degrees of freedom of the 3D head pose as a function of time

case, the 3D face model (a cloud of 3D points) is manually selected in the first stereo frame which is captured in a frontal view (Fig. 9b). This 3D face model contains about 20,500 3D points. More elaborated statistical techniques could be used for extracting the 3D facial cloud in the first range image (e.g., [18]). For subsequent frames, the registration is performed automatically by the monocular tracker and the iterative closest point algorithm (see below).

As mentioned before, the proposed evaluation consists in computing the 3D face motions using two different methods:

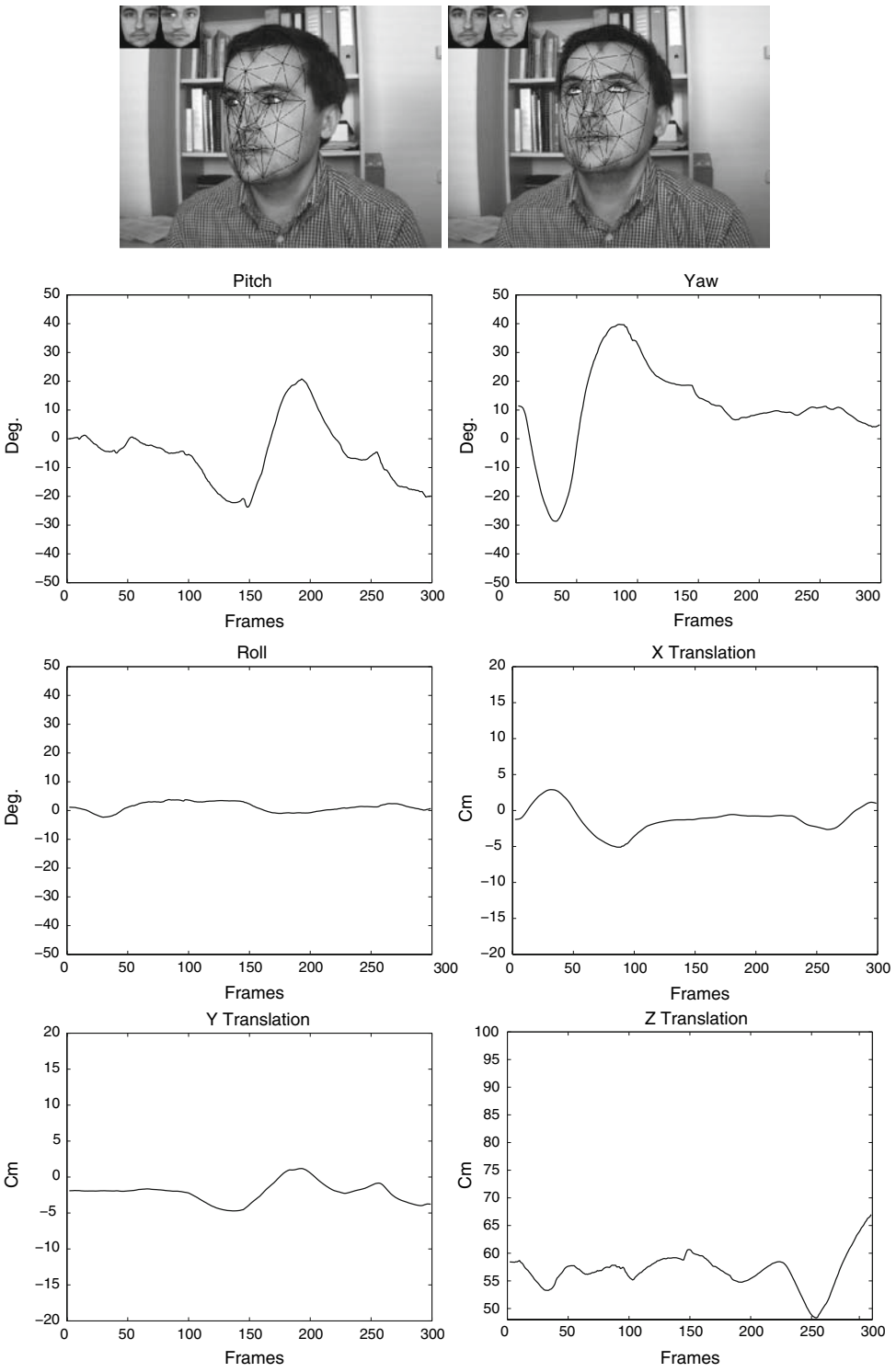

Fig. 7 The estimated 3D head pose parameters associated with the second video. Only frames 73, and 112 are shown

(1) the proposed appearance-based approach (Sect. 4) using the monocular sequence provided by the right camera, and (2) the 3D face motions computed by 3D registration of 3D facial data in different frames. Recall that the 3D rigid displacement that align two facial clouds obtained at two different frames is equivalent to the performed 3D head motion between these two frames.

The 3D registration is computed by means of the well-known iterative closest point, ICP, algorithm. ICP, also referenced in the literature as a fine registration technique, assumes

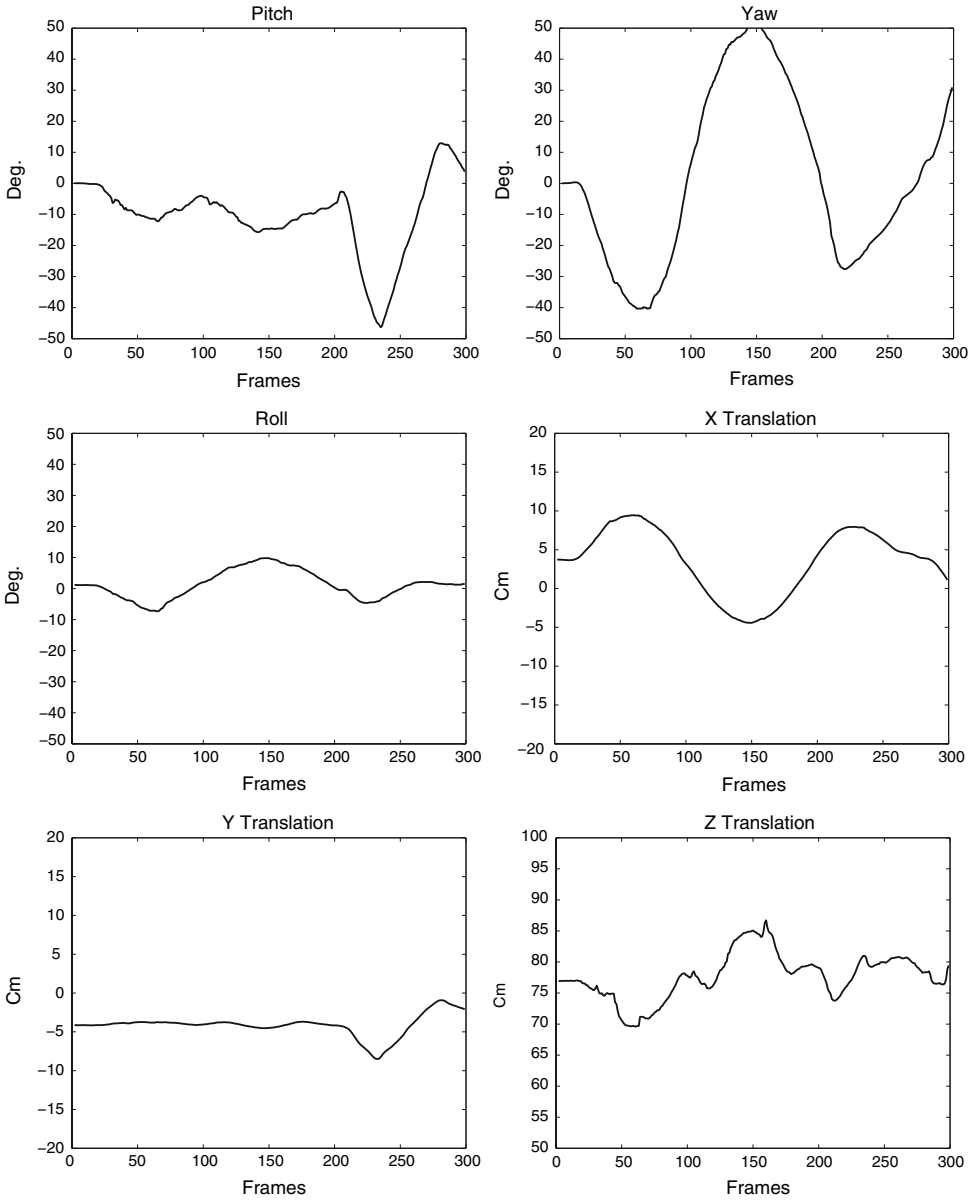

Fig. 8 The estimated 3D head pose parameters associated with the third video. Only frames 38, 167, 247, and 283 are shown

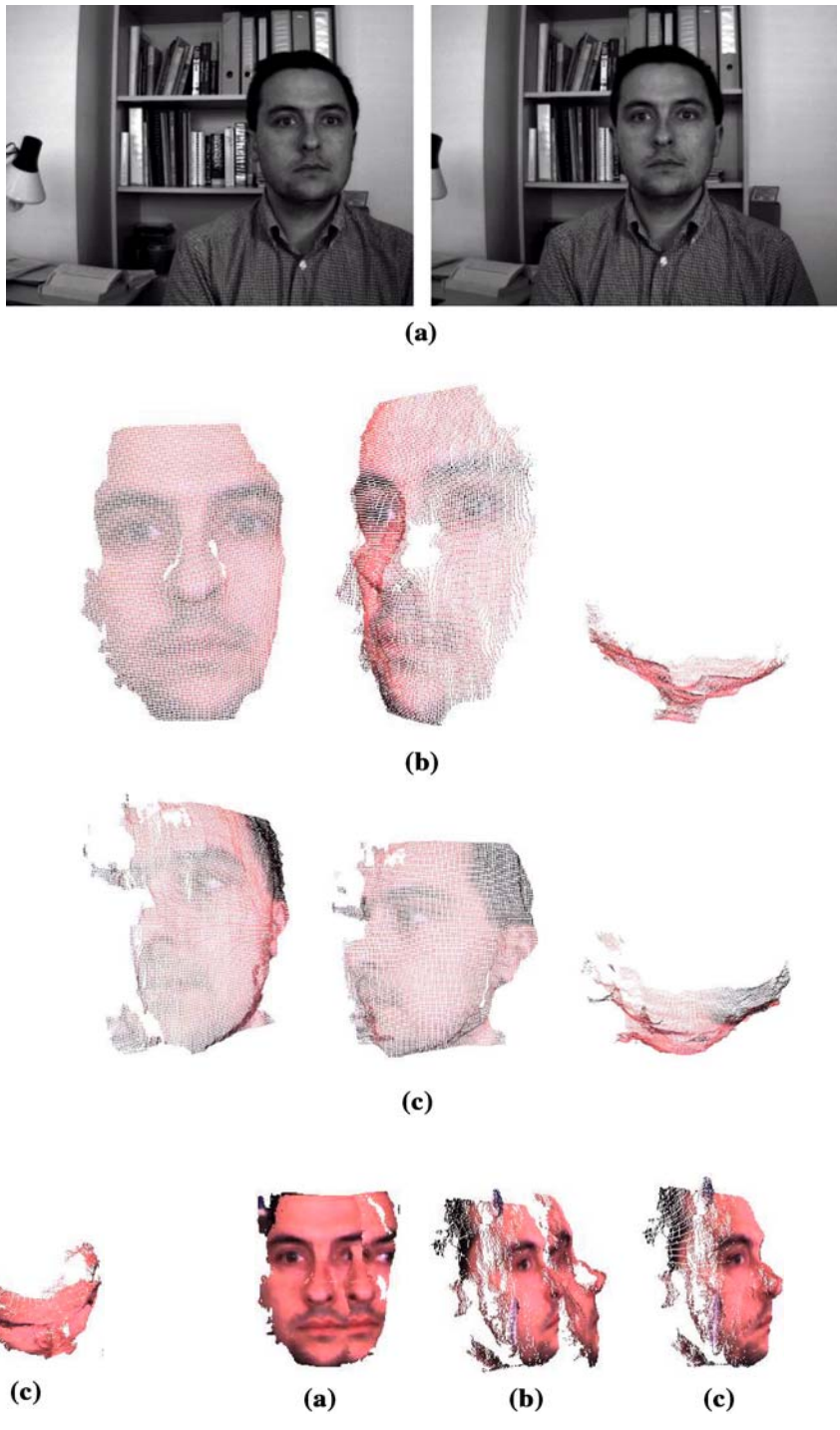

Fig. 9 Dense 3D facial data provided by a stereo head. **a** A stereo pair. **b** The corresponding computed 3D facial data with mapped texture displayed from three different points of view. **c** 3D facial data associated to another stereo pair illustrating a non-frontal face

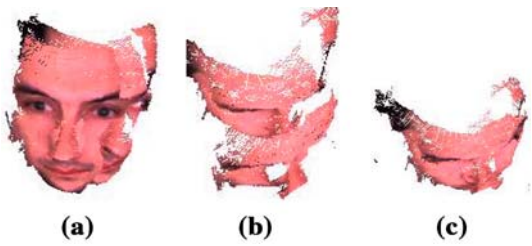

Fig. 10 3D registration of two facial clouds provided at frames 1 and 39, which are separated by a large yaw angle (about 40°). **a** The range facial data associated to frames 1 and 39, expressed in the same coordinate system. **b** Alignment results using the relative 3D face motion provided by our monocular tracker. **c** Refinement of the registration using the iterative closet point algorithm

Fig. 11 3D registration of two facial clouds provided at frames 1 and 85. **a** The range facial data associated to frames 1 and 85, expressed in the same coordinate system. **b** Alignment results using the relative 3D face motion provided by our monocular tracker. **c** Refinement of the registration using the iterative closet point algorithm

that the clouds to be registered are very close. ICP has been originally presented by Besl and McKay [3]. In our evaluation, since we use the 3D facial data/cloud in the first reference frame as a face model, the 3D registration may fail in subsequent frames containing large rotations: thus our idea is to use the monocular tracker solution as a starting solution for the ICP algorithm (see Figs. 10, 11). Therefore, the ICP

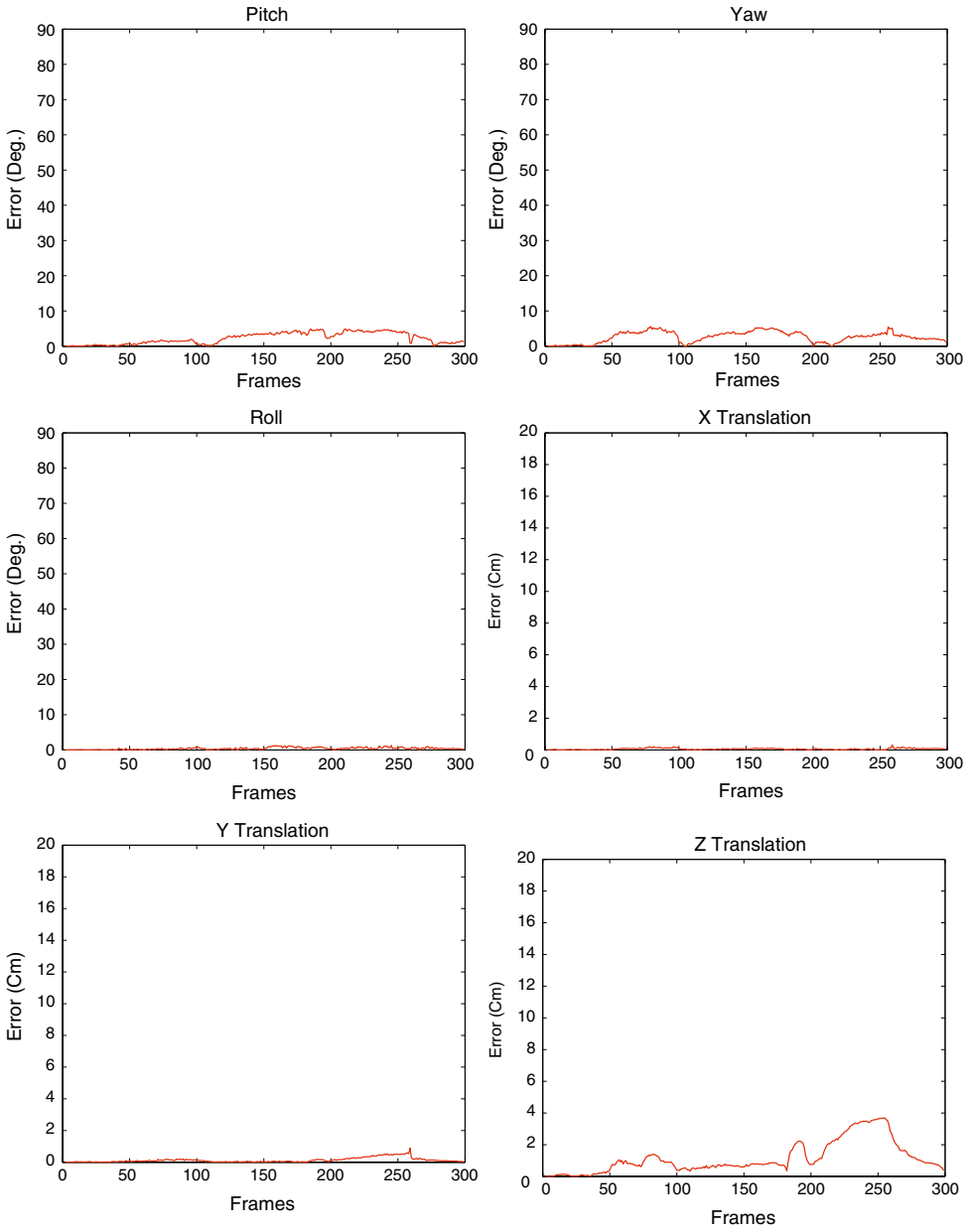

Fig. 12 3D head pose errors computed by the ICP algorithm associated with the first sequence. For each degree of freedom, the absolute value of the error is plotted. For each frame, the ICP algorithm was initialized by the output of the monocular tracker, thus the refined 3D registration can be considered as the monocular tracker error

returns a 3D rigid displacement that directly quantifies the monocular tracker accuracy.

5.2 Tracker accuracy

In order to evaluate the accuracy of the 3D head pose provided by the monocular tracker, we have used three different 300-frame stereo sequences associated with two persons. The corresponding monocular sequences are shown in Figs. 6, 7, and 8. In all these sequences, the subject was asked to move his head such that it performs the three out-of-plane motions (pitch, yaw and depth). Although the 3D head pose parameters shown in these figures are not error free (they are estimated by the monocular tracker), they give a good idea about the actual performed head motions. Thus, in the first two sequences the subjects were at about 65 cm from the cameras, in the third sequence the subject was at about 80 cm from the cameras. Moreover, in the third sequence, the vertical rotations of the head were very large, i.e., the actual yaw angle was greater than 60°.

For each stereo sequence, the 3D head pose was tracked using two approaches: (1) the monocular tracker and (2) the joint use of the stereo-based facial data and the ICP algorithm. The 3D head pose parameters provided by the monocular tracker gives the position and orientation of the 3D wireframe model with respect to the right camera frame (the one used by the monocular tracker). The 3D head motion is set to the 3D motion between the first frame and the

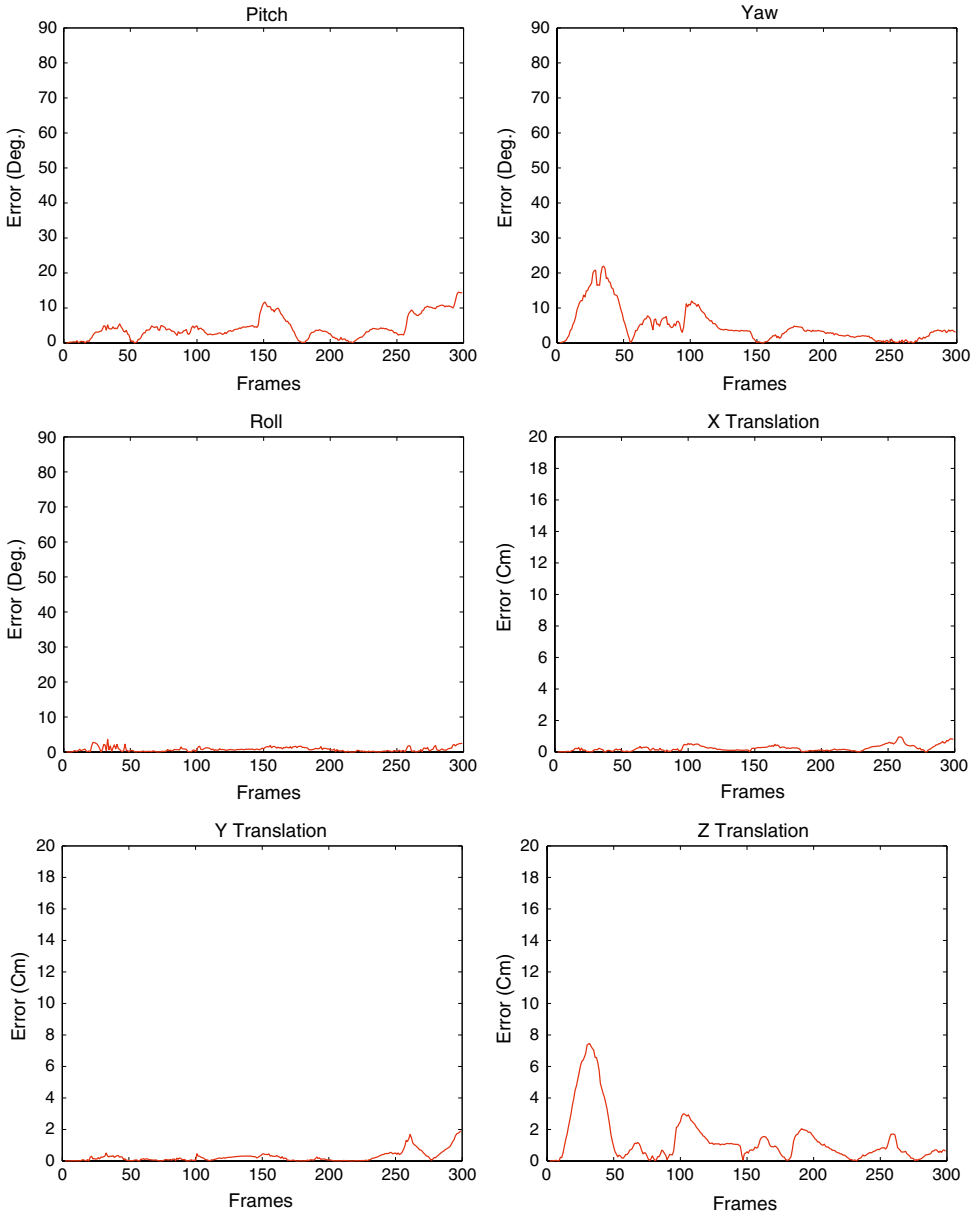

Fig. 13 3D head pose errors computed by the ICP algorithm associated with the second sequence

current frame, which is easily recovered from the corresponding absolute 3D head poses.

Figure 10 illustrates the estimated 3D head pose/motion associated with frame 39. Figure 10a illustrates the range facial data associated with frames 1 and 39, expressed in the same coordinate system. One can easily see that the face has performed a vertical rotation of about 40°. Figure 10b displays the 3D alignment obtained using the relative 3D face motion provided by the monocular tracker. Figure 10c shows the refinement of the registration using the ICP algorithm whose returned displacement gives the 3D error associated with the monocular tracker. One can notice that the monocular registration technique brings the two 3D clouds into a good alignment even though there is an offset in the in-depth translation. This is due to the monocular vision effect and to the use of a frontal texture model.

Figure 11 illustrates the estimated 3D head pose/motion associated with frame 85 using the monocular tracker and the ICP algorithm.

Figures 12 and 13 depict the monocular 3D tracker errors associated with the first and second video sequences, respectively. These are computed by the ICP algorithm as a refinement 3D displacement between two clouds. As can be seen, the 3D errors are generally quite small. One can notice the large yaw and depth errors associated with the start of the second sequence. This can be explained by the fact that the

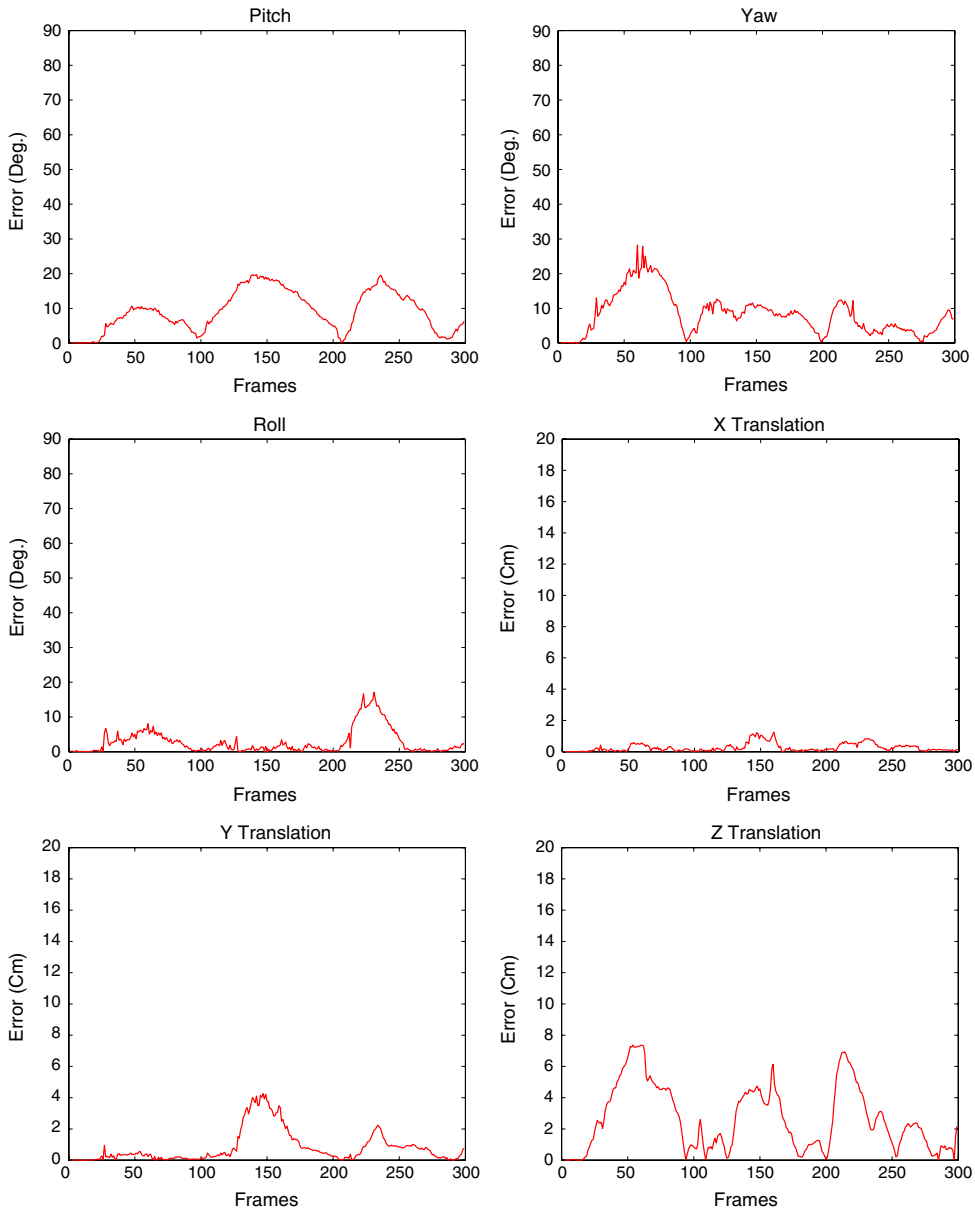

Fig. 14 3D head pose errors associated with the third sequence (see Fig. 8). For each degree of freedom, the absolute value of the error is plotted

adaptive appearance model was not very stable due the fast face motion (recall that the appearance model is built online from shape-free facial patches).

Figure 14 depicts the monocular tracker errors associated with the third sequence (depicted in Fig. 8). These errors are computed by the ICP algorithm. As can be seen, the errors increase as the face shies away from the frontal view. However, the tracker never loses the track. As can be seen, due to the effect of monocular vision and to the large out-of-plane rotations (more than 60°) the estimated depth may suffer from a 6 cm error. However, within a useful working range about the frontal view, this error is about 3 cm which corresponds to one pixel error given our camera parameters and the actual depth of the face.

Since the origin of the *Candide* model coordinate system is located on the nose's bridge and since the 3D pose errors are associated with the relative 3D motion, it follows that the translational part of the 3D motion is coupled to the rotational part. This means that an error affecting one degree of freedom could affect other degrees of freedom in order to obtain a good facial texture registration. For example, this kind of coupling is shown for frames 100–200 in Fig. 14, one can easily see that the error on the pitch angle and the in-depth translation have also affected the vertical translation.

Table 1 The average 3D head pose errors associated with the three video sequences used in the evaluation experiments

	θ_x(deg)	θ_y(deg)	θ_z(deg)	t_x(cm)	t_y(cm)	t_z(cm)
First experiment	2.24	2.63	0.40	0.06	0.13	1.15
Second experiment	4.37	4.82	0.73	0.24	0.26	1.33
Third experiment	9.17	8.28	2.79	0.26	0.81	3.34

Table 1 gives the average errors in the 3D head pose parameters associated with the three used experiments. As can be seen, the error associated with the third experiment is larger than the one associated with the first two experiments.

6 Discussion

In this paper, we have described our appearance-based 3D face tracker. We have introduced a general and efficient evaluation framework that is based on stereo range facial data. This framework can be used for evaluating any appearance-based face tracker. Moreover, it has the advantage that there is no active sensor involved.

The evaluation of the adaptive appearance-based 3D face tracker has indicated that the out-of-plane motions can be off the track whenever the absolute orientation of the face is so far from the frontal view, e.g, a vertical rotation of 60°. However, even in the extreme cases, the appearance-based tracker is still usable and does not suffer from drifting due to these out-of-plane motion inaccuracies that can be explained not only by the monocular effect but also by the fact that the texture/appearance of the 3D wireframe is modeled in a frontal view. Adopting multi-view shape-free texture models which are associated with different view points is expected to considerably decrease such inaccuracies. Alternatively, one can adopt a multi-camera system that partitions the 3D space such that at least one camera includes a near frontal view. In our case, the latter alternative could be preferred to the former one since our 3D mesh model essentially depicts a frontal face.

Although the joint use of 3D facial data and the ICP algorithm as a 3D head tracker could be attractive, the significant computational cost of the ICP algorithm prohibits real-time performance. In light of this evaluation, one is able to adjust the experimental set-up such that the monocular tracker provides accurate results. Thus, in order to obtain accurate tracking results using the monocular tracker it is always recommended to use a camera having a long focal length. However, if the camera has a short focal length or it does not have a zooming mechanism the user should be as close as possible to the camera in order to get the most accurate 3D head motions.

Acknowledgments This work was supported in part by the MEC project TRA2004-06702/AUT and The Ramón y Cajal Program.

References

1. Ahlberg, J.: An active model for facial feature tracking. EURASIP J. Appl. Signal Proc. **2002**(6), 566–571 (2002)
2. Ahlberg, J.: Model-based coding: extraction, coding, and evaluation of face model parameters. PhD thesis, No. 761, Linköping University, Sweden (2002)
3. Besl, P., McKay, N.: A method for registration of 3-D shapes. IEEE Trans. Pattern Anal. Machine Intell. **14**(2), 239–256 (1992)
4. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3D faces. In: International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH'99, pp. 187–194 (1999)
5. Blanz, V., Vetter, T.: Face recognition based on fitting a 3D morphable model. IEEE Trans. Pattern Anal. Machine Intell. 1–12 (2003)
6. Cai, Q., Aggarwal, J.K.: Tracking human motion in structured environments using a distributed-camera system. IEEE Trans. Pattern Anal. Machine Intell. **21**(12), 1241–1247 (1999)
7. Cascia, M.L., Sclaroff, S., Athitsos, V.: Fast, reliable head tracking under varying illumination: an approach based on registration of texture-mapped 3D models. IEEE Trans. Pattern Anal. Machine Intell. **22**(4), 322–336 (2000)
8. Dornaika, F., Ahlberg, J.: Fast and reliable active appearance model search for 3D face tracking. IEEE Trans. Systems Man Cybernet. Part B **34**(4), 1838–1853 (2004)
9. Dornaika, F., Davoine, F.: Head and facial animation tracking using appearance-adaptive models and particle filters. In: IEEE Workshop on Real-Time Vision for Human–Computer Interaction, Washington DC, pp. 153–162 (2004)
10. Dornaika, F., Davoine, F.: Simultaneous facial action tracking and expression recognition using a particle filter. In: IEEE International Conference on Computer Vision, pp. 1733–1738 (2005)
11. Forster, F., Lang, M., Radic, B.: Real-time 3D and color camera. In: Proceedings of the International Conference on Augmented, Virtual Environments and Three-Dimensional Imaging, pp. 45–48 (2001)
12. Gokturk, S.B., Bouguet, J.Y., Grzeszczuk, R.: A data-driven model for monocular face tracking. In: IEEE International Conference on Computer Vision, pp. 701–708 (2001)
13. Harville, M., Rahimi, A., Darell, T., Gordon, G., Woodfill, J.: 3D pose tracking with linear depth and brightness constraints. In: IEEE International Conference on Computer Vision, pp. 206–213 (1999)
14. Jebara, T.S., Pentland, A.: Parameterized structure from motion for 3D adaptative feedback tracking of faces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 144–150 (1997)
15. Jepson, A.D., Fleet, D.J., El-Maraghi, T.F.: Robust online appearance models for visual tracking. IEEE Trans. Pattern Anal. Machine Intell. **25**(10), 1296–1311 (2003)
16. Lee, D.: Effective Gaussian mixture learning for video background subtraction. IEEE Trans. Pattern Anal. Machine Intell. **27**(5), 827–832 (2005)
17. Lee, L., Romano, R., Stein, G.: Monitoring activities from multiple video streams: establishing a common coordinate frame. IEEE Trans. Pattern Anal. Machine Intell. **22**(8), 758–767 (2000)
18. Malassiotis, S., Strintzis, M.G.: Robust real-time 3D head pose estimation from range data. Pattern Recogn. **38**(8), 1153–1165 (2005)

19. Matthews, I., Baker, S.: Active appearance models revisited. Int. J. Computer Vision **60**(2), 135–164 (2004)
20. Moreno, F., Tarrida, A., Andrade-Cetto, J., Sanfeliu, A.: 3D real-time tracking fusing color histograms and stereovision. In: IEEE International Conference on Pattern Recognition, pp. 368–371 (2002)
21. Mouse, L.D.: Acoustic tracking system. http://www.vrdepot.com/vrteclg.htm
22. Proesmans, M., Gool, L.V., Oosterlinck, A.: Active acquisition of 3D shape for moving objects. In: Proceedings of the IEEE International Conference on Image Processing, pp. 647–650 (1996)
23. Yang, M.H., Kriegman, D.J., Ahuja, N.: Detecting faces in images: A survey. IEEE Trans. Pattern Anal. Machine Intell. **24**(1), 34–58 (2002)
24. Zhou, S., Chellappa, R., Mogghaddam, B.: Visual tracking and recognition using appearance-adaptive models in particle filters. IEEE Trans. Image Proc. **13**(11), 1473–1490 (2004)