FACIAL POSE AND ACTION TRACKING USING SIFT

B. H. Pawan Prasad and R. Aravind
Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai, India

Keywords: Candide, SIFT, Triangulation, 2D−3D correspondences, Pose parameters, Animation parameters.

Abstract: In this paper, a robust method to estimate the head pose and facial actions in uncalibrated monocular video sequences is described. We do not assume the knowledge of the camera parameters unlike most other methods. The face is modelled in 3D using the Candide-3 face model. A simple graphical user interface is designed to initialize the tracking algorithm. Tracking of facial feature points is achieved using a novel SIFT-based point tracking algorithm. The head pose is estimated using the POSIT algorithm in a RANSAC framework. The animation parameter vector is computed in an optimization procedure. The proposed algorithm is tested on two standard data sets. The qualitative and quantitative analysis is similar to the analysis of competing methods reported in literature. Experimental results validates that, the proposed system accurately estimates the pose and the facial actions. The proposed system can also be used for facial expression classification and facial animation.

1 INTRODUCTION

Facial pose and action tracking over the years has become an important topic in computer vision. The human head can rotate in three degrees of freedom of yaw, pitch and roll. 3D tracking of human head yields much more information compared to 2D tracking. The facial action tracking serves as an essential prerequisite for several applications such as facial expression recognition and model based image coding. It is also very useful in human computer interaction and several biometric applications.

Face tracking in 3D can be classified into two classes namely feature based and model based. The former uses the positions of local distinctive features such as eyes, mouth corners to estimate the pose (Vatahska et al., 2009). The latter uses a 3D model of the face for tracking (Dornaika and Ahlberg, 2006). Model based approaches are preferred over feature based approaches when facial action tracking is desired. In model based approaches, the model vertices are tracked frame by frame by point tracking algorithms such as Lucas-Kanade optical flow as in (Terissi et al., 2010) or normalised cross correlation as in (Dornaika and Ahlberg, 2004). The position of the model vertices directly gives the 2D−3D correspondences which are used to estimate the head pose. Accurate tracking of model points is a prerequisite for facial action tracking. More robust techniques such as the SIFT (Lowe, 2004) can be employed for model based point tracking. However, SIFT matching points between successive frames do not typically coincide with the projected model points. Hence an interpolation strategy is required to estimate the locations of the projected model points.

In this paper, we present a novel SIFT-based facial pose and action tracking algorithm capable of recovering the pose and action parameters in monocular video sequences with unknown camera parameters. SIFT is applied to successive image frames to compute matching points. Triangulation of these points is performed to estimate the positions of the model vertices by interpolation. Estimation of pose from a set of 2D−3D points is achieved using the POSIT (DeMenthon and Davis, 1995) algorithm in a RANSAC (Fischler and Bolles, 1981) framework. Once the pose for the current frame is estimated, the animation parameters are estimated in an optimization procedure.

The rest of the paper contains the following. Section 2 describes the 3D model used in our work. Tracking of facial feature points is described in Section 3. Details of facial pose and action tracking are given in Section 4. Experimental results are presented in Section 5. Finally conclusions are drawn in Section 6.

H. Pawan Prasad B. and Aravind R..
FACIAL POSE AND ACTION TRACKING USING SIFT.
DOI: 10.5220/0003362606140619
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2011), pages 614-619
ISBN: 978-989-8425-47-8

2 DEFORMABLE FACE MODEL

2.1 Modelling the Face in 3D

We use the Candide-3 face mask to model the face in 3D (Ahlberg, 2001). We have chosen six animation parameters $\boldsymbol{\alpha}$ namely, measures of jaw drop, lip stretch, lip corner lowering, raise of upper lip, eyebrow lowering and raise of outer eyebrow (Ekman and Friesen, 1977). It also consists of 14 shape parameters $\boldsymbol{\sigma}$. The model can be approximated by a linear relation given by (Ahlberg, 2001)

$$\mathbf{f}(\boldsymbol{\sigma},\boldsymbol{\alpha}) = \mathbf{g} + S\boldsymbol{\sigma} + A\boldsymbol{\alpha} \tag{1}$$

where, $\mathbf{f}$ represents the adapted face model that consists of N 3D coordinates $\mathbf{X}_i^o$ in object coordinate system concatenated into a single vector. Since for a given person, $\boldsymbol{\sigma}$ remains constant, we can write the state of the 3D model as $\mathbf{c} = [\theta_y\ \theta_r\ \theta_p\ t_x\ t_y\ t_z\ \boldsymbol{\alpha}^T]^T$, where θ_y, θ_r and θ_p represent the Euler angles of yaw, roll and pitch and $\mathbf{t} = [t_x\ t_y\ t_z]^T$ represent the translation vector. This set of six parameters constitute the 3D head pose $\mathbf{b}$.

2.2 Perspective Projection Model

The camera coordinates of the Candide-3 model vertices are obtained as $\mathbf{X}_i^c = R\mathbf{X}_i^o + \mathbf{t}$, where R is the rotation matrix and $\mathbf{t}$ is the translation vector. In our work, the camera parameters are assumed to be unknown. If we assume that the depth variations in the object are small compared to its distance from the camera, the image coordinates (x_i, y_i) are obtained by weak perspective projection as

$$x_i \approx \frac{fX_i^c}{Z^c} \quad , \quad y_i \approx \frac{fY_i^c}{Z^c} \tag{2}$$

where, f is the focal length, Z^c is the distance between any one point on the face mesh and the camera origin. The estimation of R and $\mathbf{t}$ using Eqn. 2 is robust to errors in the choice of f (Aggarwal et al., 2005).

2.3 Initialization

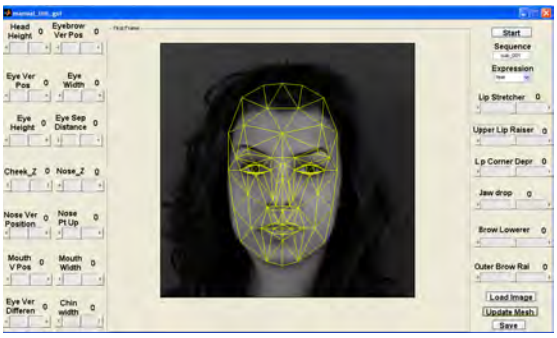

Figure 1: Face Mesh Initialization.

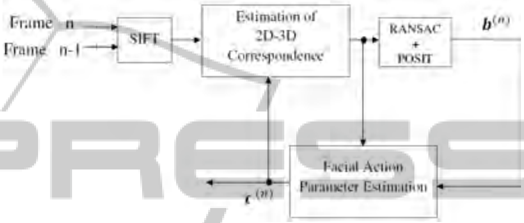
Frame n
Frame n-1
SIFT
Estimation of 2D-3D Correspondence
RANSAC + POSIT
$b^{(n)}$
Facial Action Parameter Estimation
$c^{(n)}$

Figure 2: Block Diagram of the Proposed System.

During initialization, we assume that the face shows a frontal view with zero Euler angles. The initial values of $\boldsymbol{\alpha}^{(0)}$ and $\boldsymbol{\sigma}^{(0)}$ are computed manually using the a Graphical User Interface as shown in Figure 1. The adapted face model $\{\mathbf{X}_i^f\}_{i=1}^N = (X_i^f, Y_i^f, Z_i^f)$ is computed by first estimating the horizontal and the vertical scaling factors h and v, which determine the amount by which the face mesh has to be scaled in the x and the y dimensions respectively. The two eye corners (m_1, n_1) and (m_2, n_2) and one mouth corner (m_3, n_3) are marked on the given image. The corresponding points (x_1, y_1), (x_2, y_2) and (x_3, y_3) on the projected face mesh are also selected. Hence $h = \frac{|x_1 - x_2|}{|m_1 - m_2|}$, $v = \frac{|y_1 - y_3|}{|n_1 - n_3|}$. The horizontal and vertical translations d_x and d_y are computed using an eye corner. Finally the adapted face model is obtained using inverse perspective projection as

$$\begin{aligned} X_i^f &= \left(\frac{Z^c + Z_i^o}{f}\right)(x_i^s - d_x) \quad ; Z_i^f = dZ_i^o \\ Y_i^f &= \left(\frac{Z^c + Z_i^o}{f}\right)(y_i^s - d_y) \quad ; i = 1, 2, \ldots N \end{aligned} \tag{3}$$

We have set $f = 1000$ and $Z^c = 50000$ which was determined by experimental evaluations. We have chosen the value of the scaling factor d as one percent of the distance from the camera Z^c, to make sure the depth variations in the object is small compared to the distance from the camera. Once these values are set, we can use the same values for estimating head pose and action parameters in any other video sequence.

3 TRACKING FEATURE POINTS

The block diagram of the proposed tracking system is shown in Fig. 2. The head pose and facial actions are decoupled and estimated in two different stages as proposed in (Dornaika and Ahlberg, 2004). The first stage consist of global adaptation where, the 3D head pose parameters are estimated. In the second stage, local adaptation is performed to estimate the animation parameters.

3.1 Scale Invariant Feature Transform

Scale Invariant Feature Transform (SIFT) (Lowe, 2004) is a method for extracting distinctive image features from images that are invariant to scale and rotation, change in illumination, and also across a limited change in 3D viewpoint. Given two successive frames of a video sequence $n-1$ and n, applying SIFT we obtain a set of P matching points $\{\mathbf{p}_i\}_{i=1}^{P}$ and $\{\mathbf{q}_i\}_{i=1}^{P}$ between frames $n-1$ and n respectively.

We are interested in determining only the $M \leq P$ points $\{\mathbf{m}_i\}_{i=1}^{M}$ and $\{\mathbf{n}_i\}_{i=1}^{M}$ that are present inside the face region. Given the object state $\mathbf{c}^{(n-1)}$ at frame $n-1$, the adapted model is then rotated and translated to get the 3D coordinates of the model for the frame $n-1$.

$$\mathbf{X}_i^{(n-1)} = R^{(n-1)}\mathbf{X}_i^{f} + \mathbf{t}^{(n-1)} \tag{4}$$

The convex hull obtained by projecting these 3D points on to the image plane forms a polygon. The SIFT points $\{\mathbf{m}_i\}_{i=1}^{M}$ that are present inside this polygon are determined using the fundamental point-location problem of computational geometry (De Berg et al., 2008).

3.2 Tracking Mesh Vertices

The goal here is to determine the location of the mesh vertices $\{\mathbf{x}_i\}_{i=1}^{N}$ at frame n. However, normalised cross correlation as in (Jang and Kanade, 2008) or nearest neighbour matching translation as in (Brox et al., 2010) was possible if a cylindrical face model with densely lying points was used. Since we use the Candide-3 face model, both these methods leads to accumulation of errors. Hence we propose an interpolation strategy to determine the location of the mesh vertices using the pose $\mathbf{b}^{(n-1)}$ of the frame $n-1$ and the SIFT matching points.

The M SIFT points inside the face region form a new convex hull which encloses K out of the N mesh points denoted as $\{\mathbf{u}_i\}_{i=1}^{K}$. The corresponding 3D points from the adapted face model are denoted as $\{\mathbf{U}_i^{f}\}_{i=1}^{K}$. The K mesh points $\{\mathbf{u}_i\}_{i=1}^{K}$ of frame $n-1$ have one-to-one correspondence with K points $\{\mathbf{v}_i\}_{i=1}^{K}$ in frame n, the locations of which are unknown because, the pose at frame n is unknown. The M SIFT points are now connected using Delaunay triangulation (Edelsbrunner, 2001). Let us consider a mesh point $\mathbf{u}_i$ which is enclosed by a triangle formed by three vertices from the set $\{\mathbf{m}_i\}_{i=1}^{M}$. In order to estimate the location of $\mathbf{v}_i$, we first compute the barycentric coordinates of $\mathbf{u}_i = (u_i^x,\ u_i^y)$ denoted as $\Lambda_i = [\lambda_{i,1}\ \ \lambda_{i,2}]^T$. We obtain the vertices of the triangle inside which $\mathbf{u}_i$ lies as say, $\mathbf{r}_i = [r_i^x\ \ r_i^y]^T$, $\mathbf{s}_i = [s_i^x\ \ s_i^y]^T$ and $\mathbf{t}_i = [t_i^x\ \ t_i^y]^T$. Hence $\mathbf{u}_i$ can be written as a weighted sum of these three vertices (Bradley, 2007) as

$$\begin{aligned}
\mathbf{u}_i &= \lambda_{i,1}\mathbf{r}_i + \lambda_{i,2}\mathbf{s}_i + (1-\lambda_{i,1}-\lambda_{i,2})\mathbf{t}_i \\
&= \begin{bmatrix} r_i^x - t_i^x & s_i^x - t_i^x \\ r_i^y - t_i^y & s_i^y - t_i^y \end{bmatrix} \begin{bmatrix} \lambda_{i,1} \\ \lambda_{i,2} \end{bmatrix} + \begin{bmatrix} t_i^x \\ t_i^y \end{bmatrix} \\
\mathbf{u}_i &= \mathscr{A}_i \Lambda_i + \mathbf{t}_i \qquad 1 \leq i \leq K && (5) \\
\Rightarrow \Lambda_i &= (\mathscr{A}_i)^{-1}(\mathbf{u}_i - \mathbf{t}_i) && (6)
\end{aligned}$$

Let us denote the three points at frame n that have SIFT correspondence to the triangle vertices $\mathbf{r}_i$, $\mathbf{s}_i$ and $\mathbf{t}_i$ as $\mathbf{w}_i$, $\mathbf{y}_i$ and $\mathbf{z}_i$ respectively. Here we assume that $\mathbf{v}_i$ lies inside the triangle formed by the three vertices $\mathbf{w}_i$, $\mathbf{y}_i$ and $\mathbf{z}_i$. Hence we can write $\mathbf{v}_i$ as a weighted sum of these three vertices as before. To estimate $\mathbf{v}_i$ at frame n, we use the same barycentric coordinates Λ_i of $\mathbf{u}_i$ computed at frame $n-1$. Therefore,

$$\begin{aligned}
\mathbf{v}_i &= \mathscr{B}_i \Lambda_i + \mathbf{z}_i \qquad 1 \leq i \leq K \\
&= \mathscr{B}_i (\mathscr{A}_i)^{-1}(\mathbf{u}_i - \mathbf{t}_i) + \mathbf{z}_i && (7)
\end{aligned}$$

where $\mathscr{B}_i$ is computed as in Eqn. 5 using the vertices $\mathbf{w}_i$, $\mathbf{y}_i$ and $\mathbf{z}_i$ in place of $\mathbf{r}_i$, $\mathbf{s}_i$ and $\mathbf{t}_i$. Eqn. 7 holds true if the following three constraints are satisfied. Firstly,the SIFT matching points are accurate. Secondly, the face does not undergo any local deformation caused due to facial appearance changes. Thirdly, the head pose at frame $n-1$ is precisely known. Any one of the above constraints not holding true, leads to the occurrence of outliers. Handling of outliers is discussed in sections that follow. The set of $2D-3D$ correspondences $\{\mathbf{v}_i\}_{i=1}^{K}$ and $\{\mathbf{U}_i^{f}\}_{i=1}^{K}$ is used to determine the pose at frame n as described next.

4 ESTIMATION OF FACIAL POSE AND ACTION PARAMETERS

Once the $2D-3D$ correspondences are established, facial pose is estimated as described in our earlier work (Pawan and Aravind, 2010) which makes use of the adaptation strategy proposed in (Dornaika and Ahlberg, 2004). In this section, we develop an algorithm to estimate the facial animation parameter vector $\boldsymbol{\alpha}$ associated with the current frame given the knowledge of the head pose parameter vector $\mathbf{b}$ and a set of K mesh vertices $\mathbf{v}_i$ estimated in Section 3.2.

The animation parameter vector $\boldsymbol{\alpha}$ consists of six parameters, four of which namely the measures of jaw drop, lip stretch, lip corner lowering, raise of upper lip modify the position of mouth and jaw in the lower face denoted as $\boldsymbol{\alpha}_l$. The other two parameters, namely the measures of eyebrow lowering and raise of outer eyebrow modify the position of eyebrow in the upper face denoted as $\boldsymbol{\alpha}_u$. The corresponding animation parameter matrices are denoted as A_l and A_u. The Candide-3 face model consists of N 3D vertices. We denote a subset of these N vertices that are related to the facial actions in the upper face as $\mathcal{F}_u$ and lower face as $\mathcal{F}_l$ that are mutually exclusive (Ahlberg, 2001). The upper face model can be represented by a vector $\mathbf{g}_u$ obtained by concatenating all the N_u 3D vertices. The lower face model denoted by vector $\mathbf{g}_l$ of dimension N_l is similarly computed. We then pick $3N_u$ points from the vector $S\boldsymbol{\sigma}$ that correspond to the upper face and denote it as $\mathbf{s}_u$ and $\mathbf{s}_l$ is computed similarly. From Eqn. 1 we can write,

$$\begin{aligned} \mathbf{f}_u(\boldsymbol{\sigma},\boldsymbol{\alpha}_u) &= \mathbf{g}_u + \mathbf{s}_u + A_u\boldsymbol{\alpha}_u \\ \mathbf{f}_l(\boldsymbol{\sigma},\boldsymbol{\alpha}_l) &= \mathbf{g}_l + \mathbf{s}_l + A_l\boldsymbol{\alpha}_l \end{aligned} \tag{8}$$

The initialization parameters such as scaling factors h, v, d and the translations d_x, d_y computed in Section 2.3 are then incorporated into the face model in a similar way as in Eqn. 3 to get the vectors $\widehat{\mathbf{f}}_u$ and $\widehat{\mathbf{f}}_l$. The adapted face model is then rotated and translated to give

$$\begin{aligned} \mathbf{k}_u(\boldsymbol{\sigma},\boldsymbol{\alpha}_u,\mathbf{b}) &= \mathbb{R}_u\widehat{\mathbf{f}}_u + \widetilde{\mathbf{t}}_u \\ \mathbf{k}_l(\boldsymbol{\sigma},\boldsymbol{\alpha}_l,\mathbf{b}) &= \mathbb{R}_l\widehat{\mathbf{f}}_l + \widetilde{\mathbf{t}}_l \end{aligned} \tag{9}$$

where, $\mathbb{R}_u$ is the block diagonal rotation matrix of size $3N_u \times 3N_u$ given by $\mathbb{R}_u = \text{diag}(R,R,..,R)$, similarly $\mathbb{R}_l$ is of size $3N_l \times 3N_l$ and $\widetilde{\mathbf{t}}_u$ and $\widetilde{\mathbf{t}}_l$ are the translation vectors of dimension $3N_u$ and $3N_l$ respectively given by $\widetilde{\mathbf{t}}^T = [\mathbf{t}^T \;\; \mathbf{t}^T \ldots \;\; \mathbf{t}^T]$.

The 3D vertices of the face model present in the vectors $\mathbf{k}_u$ and $\mathbf{k}_l$ are projected onto the image plane using weak perspective projection as in Eqn. 2. We denote these 2D vertices as $\{\mathbf{h}_{u,i}\}_{i=1}^{N_u}$ and $\{\mathbf{h}_{l,i}\}_{i=1}^{N_l}$ respectively. Since, $\mathcal{F}_u$ and $\mathcal{F}_l$ are mutually exclusive, the optimization is decoupled to reduce the influence of one over the other. We define two cost functions.

$$c(\boldsymbol{\alpha}_u,\mathbf{b}) = \sum_{i\in\mathcal{F}_u} T\left(||\mathbf{v}_i - \mathbf{h}_{u,i}(\boldsymbol{\alpha}_u,\mathbf{b})||^2\right); 1 \le i \le N_u$$

$$c(\boldsymbol{\alpha}_l,\mathbf{b}) = \sum_{i\in\mathcal{F}_l} T\left(\left|\left|\mathbf{v}_i - \mathbf{h}_{l,i}(\boldsymbol{\alpha}_l,\mathbf{b})\right|\right|^2\right); 1 \le i \le N_l$$

The number of tracked facial feature points $\mathbf{v}_i$ can be less than N_u or N_l. In this case, only the available feature points are used to estimate the animation parameter vector $\boldsymbol{\alpha}$. The $T(\cdot)$ function denotes the robust Tukey bi-square M-estimator (Maronna et al., 2006). It is used to reduce the effect of outliers caused due to SIFT mismatches. We determine the optimum value of $\widehat{\boldsymbol{\alpha}}^T = [\widehat{\boldsymbol{\alpha}}_u^T \;\; \widehat{\boldsymbol{\alpha}}_l^T]$ by minimizing the following expressions.

$$\begin{aligned} \widehat{\boldsymbol{\alpha}}_u &= \underset{\boldsymbol{\alpha}_u}{\text{argmin}} \;\; c(\boldsymbol{\alpha}_u,\mathbf{b}) \\ \widehat{\boldsymbol{\alpha}}_l &= \underset{\boldsymbol{\alpha}_l}{\text{argmin}} \;\; c(\boldsymbol{\alpha}_l,\mathbf{b}) \end{aligned} \tag{10}$$

The optimum value of $\boldsymbol{\alpha}^{(n)}$ for the current frame n is computed by searching in the local neighbourhood of the estimate $\boldsymbol{\alpha}^{(n-1)}$ of the previous frame $n-1$. The optimization problem of Eqn. 10 is solved using nonlinear least squares approach (Levenberg-Marquardt) (Levenberg, 1944) (Marquardt, 1970).

5 EXPERIMENTAL RESULTS

In this section, we first show the advantage of the proposed method to estimate the location of mesh vertices over nearest neighbour translation described in (Brox et al., 2010). Then we report performance studies that evaluate the proposed head pose and facial action tracking system.

5.1 Feature Point Tracking

The SIFT matching points between successive frames do not necessarily coincide with the projected mesh vertices. If more mesh points were considered using a finer mesh, the nearest neighbour method would have been sufficient to estimate the location of mesh vertices. The nearest neighbour translation was proposed in (Brox et al., 2010) which estimates the location of mesh vertices by making the nearest mesh point to undergo the same 2D translation present between SIFT matching points. However, this method wont work when a mesh such as Candide-3 with only $N = 113$ vertices is used. The other disadvantage is in the fact that, the above method is not suitable for head pose estimation as it does not capture the 3D movement accurately because, it performs a 2D translation of the mesh vertex. This results in error accumulation and tracker loses track after a few frames. The proposed method scores well in these scenarios. To validate this argument, we present a video sequence “Vam8” from the BU head pose database. We performed two experiments, Figure 3 (a) shows the tracker output using nearest neighbour translation described in (Brox et al., 2010). Figure 3 (b) shows the tracker output using the proposed scheme. It is evident that, the tracker in the former case loses the face just after a few frames due to error accumulation. The proposed scheme is able to handle the 3D movement better. Figure 4 (a) illustrates the SIFT correspondences with estimated mesh

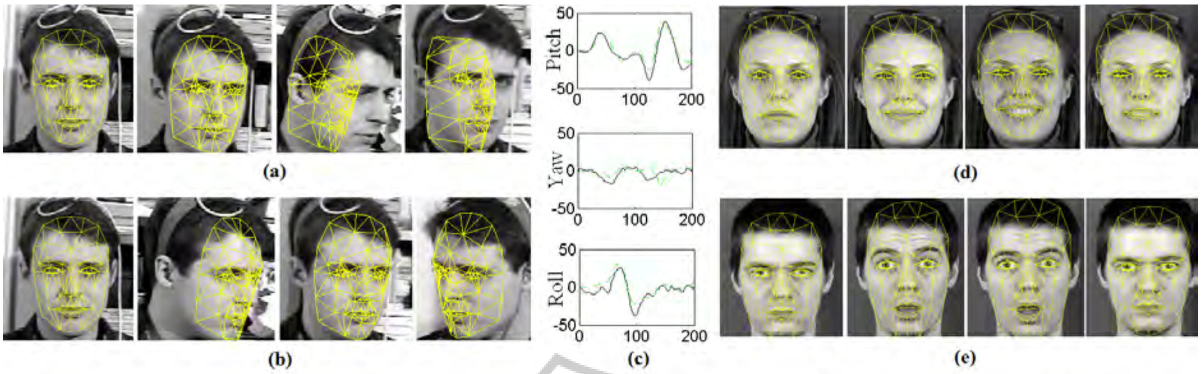

Figure 3: (a) Nearest neighbour translation described in (Brox et al., 2010), frames 1, 37, 63, 91, (b) Proposed feature point tracking algorithm, frames 1, 63, 91, 140, (c) Estimated Rotation angles for the sequence "Vam5", (d) "Happiness" sequence, Frames 1, 16, 28, 47, (e) "Surprise" sequence, Frames 1, 14, 28, 50.

vertices for the sequence "Vam5". The number of putative mesh correspondences and the respective inliers for the same sequence is shown in Figure 4 (b).

5.2 Facial Pose and Action Tracking

To evaluate the performance of the proposed facial pose and action tracking system, we use two different datasets. Firstly, the algorithm is tested on the Boston University database (La Cascia et al., 2000). It consists of 72 image sequences of 200 frames each of size 320 × 240, that contains eight people, each of them appearing in nine videos. Out of the eight sets of nine videos each, five sets were taken under uniform illumination and the rest were taken under varying illumination. The ground truth indicating the Euler angles is available for all the 72 sequences. We have tested the proposed algorithm on all the 72 video sequences to evaluate its robustness under uniform as well as varying illumination. The average mean absolute errors for roll, yaw and pitch are tabulated in Table 1. Figure 3 (c) shows the estimated Euler angles against the ground truth for the "Vam5" sequence.

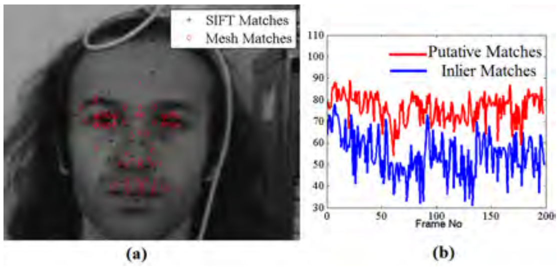

Figure 4: (a) SIFT correspondences with estimated Mesh vertices (b) No. of Mesh Vertices along with No. of Inliers for the sequence "Vam5".

Table 1: MAE of different algorithms.

Algorithm	MAE (deg)		
	Pitch	Yaw	Roll
Proposed Method	**2.5**	**3.8**	**3.6**
(Jang and Kanade, 2008)	3.7	4.6	2.1
(Xiao et al., 2003)	3.8	3.2	1.4
(Choi and Kim, 2008)	3.92	4.04	6.71

Next, we use the Multimedia Understanding Group facial expression database (Group, 2007) which consists of image sequences of 86 subjects performing six basic expressions namely, anger, disgust, fear, happiness, surprise and sad. We evaluate the performance of the proposed algorithm on a subset of this dataset containing 10 out of the 86 subjects performing six different expressions. Figure 3 (d) shows image frames of a sequence in which the subject performs happiness expression. Next we consider an image sequence in which the subject performs surprise expression as shown in Figure 3 (e). The proposed algorithm is able to successfully track the movement of lips and eyebrows. To evaluate the proposed algorithm quantitatively, ground truth for the above datasets was established using a scheme similar to manual initialization described in Section 2.3. The animation parameter vector α is recorded for every frame of the video sequence manually. Figure 5 shows the estimated animation parameters versus the ground truth for different expressions.

6 CONCLUSIONS

In this paper, we have proposed a novel technique to estimate the head pose and facial animation parameters in a monocular video sequence. The pose and animation parameters were recovered without assuming the knowledge of the internal camera parameters. The focus was more on developing a fully robust sys-

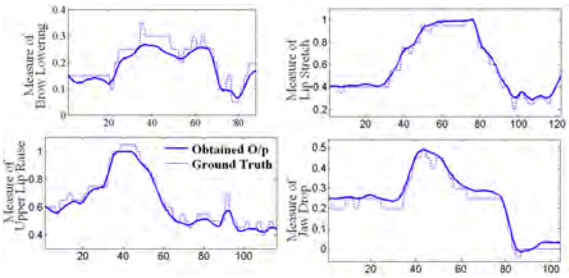

Figure 5: Animation Parameters for different sequences against the ground truth.

tem rather than achieving real time performance. The proposed system was first evaluated for its robustness on standard head pose estimation datasets. The mean absolute errors of yaw, pitch and roll were found to be comparable and in some cases better than the results reported in literature. The proposed system was next tested on a standard facial expression dataset which largely involved movements of eyebrows and mouth. Experimental results show that the proposed algorithm is able to effectively handle the mouth and brow movements. We manually collected the ground truth for several facial expression test sequences to evaluate the algorithm quantitatively. The estimated animation parameters were found to agree very well with the ground truth.

REFERENCES

Aggarwal, G., Veeraraghavan, A., and Chellappa, R. (2005). 3d Facial pose tracking in Uncalibrated videos. *PRMI*, pages 515–520.

Ahlberg, J. (2001). Candide-3–an updated parametrized face. *Report No. LiTH-ISY*.

Bradley, C. (2007). The Algebra of Geometry: Cartesian, Areal and Projective Co-ordinates. *Highperception Ltd., Bath*.

Brox, T., Rosenhahn, B., Gall, J., and Cremers, D. (2010). Combined region and motion-based 3D tracking of rigid and articulated objects. *PAMI*, 32(3):402.

Choi, S. and Kim, D. (2008). Robust head tracking using 3D ellipsoidal head model in particle filter. *Pattern Recognition*, 41(9):2901–2915.

De Berg, M., Cheong, O., Van Kreveld, M., and Overmars, M. (2008). *Computational geometry: Algorithms and applications*. Springer.

DeMenthon, D. and Davis, L. (1995). Model-based object pose in 25 lines of code. *IJCV*, 15(1):123–141.

Dornaika, F. and Ahlberg, J. (2004). Face and facial feature tracking using deformable models. *IJIG*, 4(3):499.

Dornaika, F. and Ahlberg, J. (2006). Fitting 3D face models for tracking and active appearance model training. *Image and Vision Computing*, 24(9):1010–1024.

Edelsbrunner, H. (2001). *Geometry and topology for mesh generation*. Cambridge Univ. Press.

Ekman, P. and Friesen, W. (1977). Facial Action Coding System. *Consulting Psychology Press*.

Fischler, M. and Bolles, R. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395.

Group, M. U. (2007). *The MUG Facial Expression Database*. http://mug.ee.auth.gr/fed/.

Jang, J. and Kanade, T. (2008). Robust 3D head tracking by online feature registration. In *8th IEEE International Conference on Automatic Face and Gesture Recognition*.

La Cascia, M., Sclaroff, S., and Athitsos, V. (2000). Fast, reliable head tracking under varying illumination: an approach based on registration of texture-mapped 3 D models. *PAMI*, 22(4):322–336.

Levenberg, K. (1944). A method for the solution of certain nonlinear problems in least-squares. *The Quarterly of Applied Mathematics*, 2:164–168.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110.

Maronna, R., Martin, R., and Yohai, V. (2006). *Robust statistics*. Wiley New York.

Marquardt, D. (1970). Generalized inverses, ridge regression, biased linear estimation, and nonlinear estimation. *Technometrics*, 12(3):591–612.

Pawan, P. and Aravind, R. (2010). A Robust Head Pose Estimation System in Uncalibrated Monocular Videos. In *Indian Conference on Computer Vision Graphics and Image Processing*. ACM.

Terissi, L., Gómez, J., CIFASIS, C., and Rosario, A. (2010). 3D Head Pose and Facial Expression Tracking using a Single Camera. *Journal of Universal Computer Science*, 16(6):903–920.

Vatahska, T., Bennewitz, M., and Behnke, S. (2009). Feature-based head pose estimation from images. In *7th IEEE-RAS International Conference on Humanoid Robots*, pages 330–335. IEEE.

Xiao, J., Moriyama, T., Kanade, T., and Cohn, J. (2003). Robust full-motion recovery of head by dynamic templates and re-registration techniques. *International Journal of Imaging Systems and Technology*, 13(1):85–94.