3D Facial Pose Tracking in Videos

G.Aggarwal A.Veeraraghavan R.Chellappa
University of Maryland, College Park
College Park, Md-20742

Abstract

This paper presents a method to recover the 3D configuration of a face in each frame of a video. The 3D configuration consists of the three translational parameters (recoverable upto a scale factor) and the three orientation parameters which correspond to the yaw, pitch and roll of the face. This is equivalent to maintaining 3D correspondences on a face as the face moves in a video. Maintaining such explicit correspondence makes it suitable as a front-end for applications like face modeling and recognition, facial expression analysis, lip reading, eye tracking, etc. which require head stabilization (or normalization). The approach combines the structural advantages of geometric modeling with the statistical advantages of a particle-filter based inference. The face is modeled as the curved surface of a cylinder with elliptical cross-section which is free to translate and rotate arbitrarily about its axes. The geometric modeling takes care of pose and self-occlusion while the statistical modeling handles occlusion and illumination variations. Experimental results on multiple datasets are provided to show the efficacy of the approach.

1 Introduction

Face tracking is a crucial task for several applications in computer vision. It serves as the first step in several applications like face recognition, lip reading, human computer interaction and animation. Most of these applications require that actual 3D parameters of the motion of the head, like the orientation of the head, to be recovered. In this paper, we propose an approach based on a cylindrical model for the head for reliable tracking of position and orientation of the head under illumination changes, occlusion and extreme poses. We also qualitatively show how such 3D tracking of the head improves the performance of recognition systems.

1.1 Prior Work

There has been significant work on facial tracking using 2D appearance based models. [12] [18] [21] use 2D face models based on splines or deformable templates. [1] [9] use affine and planar models, respectively to track a face. Quite clearly, such approaches based on the 2D appearances usually do not explicitly solve the correspondence problem. Rather, more often than not they are interested in finding just the image region containing the object (face in this case). Estimation of the 3D orientation of the head is extremely difficult using these approaches. Therefore when such 2D approaches are used as a front-end for tasks such as recognition, multiple view based exemplars [22] are sometimes used in the gallery. While, such a system might improve over the performance of single image based face-recognition systems, such view based exemplars do not capture the structure of the object in view (i.e., structure of a face.).

Recently, several methods have been developed for 3D face tracking. [10] uses a closed loop approach that utilizes a structure from motion algorithm to generate a 3D model of the face. The model is then used to constrain the features in the next frame. The tracking is based on a Kalman filter. In [20], techniques in continuous optimization are applied to a linear combination of 3D face models. They are able to automatically recover the face position and expression for each frame. [16] proposes a hybrid sampling solution using both RANSAC and particle filters to track the pose of a face. Some researchers have proposed using active appearance models for face tracking and/or pose recovery and expression recognition [5][13]. A cylindrical face model for face tracking has been used in [3]. In their formulation, the interframe warping function is assumed to be locally linear. In addition, they also assume that the interframe pose change occurs only in one of the six degrees of freedom of the rigid cylinderical model. In our approach, we do not have to make any such assumptions. This improves both tracking accuracy and robustness.

In this paper we propose a method for tracking facial pose in a video. In the next section we discuss the geometric modeling of the face. Section 3 presents the featues used for tracking. In Section 4 we discuss our particle filter-based tracking algorithm. Section 5 presents experiments on tracking and recognition. We present the conclusions of the work in section 6.

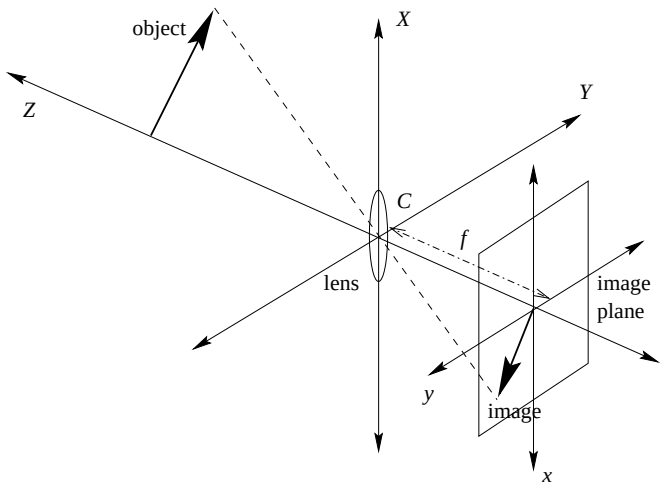


Figure 1: 3D imaging geometry.

2 The Geometric Model

The choice of the model to represent the facial structure is very crucial for the problem of face tracking. Several geometric models have been proposed for facial analysis. More often than not, the choice depends on the goal of the analysis. There are several algorithms that do not assume an explicit structural model. They track salient points, features or 2D image patches [19] to recover the 2D or 3D head configuration. On the other extreme, there are algorithms like [10] that use a set of 3D laser-scanned heads represented in a parameterized eigenspace to constrain the structural estimation. A few other focus mainly on 2D tracking (e.g., [22], [2], [4], [7], [9], [21]) which makes a planar model (elliptic, rectangular, etc.) suitable for them.

We would like to restate here that in our work, we aspire to estimate the 3D configuration of the face in each frame. Though a planar model will probably be the simplest one to use, it does not have the capability to handle out-of-plane rotations due to the involved self-occlusions. Moreover the parameters recovered using such a model does not contain information required to estimate the 3D configuration of the face. On the other hand, using a complicated face model (e.g., 3D range data model of an average face), makes the initialization and registraion process difficult. In fact, [3] shows experiments where perturbations in the model parameters affect the tracking performance using a complex rigid model (generated by averaging the Cyberware scans of several people), while the simple cylindrical model is robust to such perturbations.

Similar to [3], we use a cylindrical model, though with an elliptical cross-scetion, to represent a face. The choice of the elliptic cylinder was based on the observation that for most people, the cross section of the head is more elliptic than cylindrical. The choice of the ellipticity does not affect the tracking performance in general but it does make a difference when the face is turned about the vertical axis by a large angle (i.e., high yaw value). Assuming that our cylindrical model reasonably approximates the 3D structure of a face, the problems related to pose and self-occlusion (usually due to pose changes) get automatically taken care of.

From the point of geometrical modeling, the next important issue is the choice of projection. Due to the absence of the calibration parameters of the camera, people usually assume orthographic projection. Though orthographic assumption is usually considered to be accpetable when the object (or face in our case) is far away from the camera, there are a couple of issues in making such an assumption with regard to the cylindrical model we use. First of all, orthographic projection essentially assumes that the visible portion of the face (or object in general) is planar. Not only does it contradicts our cylindrical model but is also probably not a good approximation when the face is close to the camera which is usally the case with most facial analysis applications. Moreover, we observed that under orthographic assumption, it becomes diffcult to distinguish between scale and pitch. These reasons motivate us to use perspective projection model similar to the one shown in Figure 1.

One might feel that the use of the pespective model necessitates the need to know atleast the focal length of the camera. In our formulation, we use an arbitrary focal length and still get very good results. This can be justified as follows. Though probably counter-intuitive, it is not necessary to have knowledge either about the calibration parameters of the camera or about the scene geometry to compute the rotation involved between two frames of a video when the motion involved is pure rotation. This comes from the fact that there exist a homography between any two images of

the same scene if the underlying motion between the two camera configurations is pure rotation. Such a homography (or mapping) does not depend either on the camera parameters or the scene being viewed. From the point of our problem, this means that if we know the translation, the face has undergone, under the assumed projection model (perspective with an arbitrary focal length), we can estimate the change in the orientation of the face without actually knowing the correct focal length of the camera. Note that though the translation values will be different (upto a scale factor) in the assumed model, the rotations involved remain the same. We will revisit this issue later while describing the particle-filter framework.

2.1 Model Initialization

The model is initialized using the first frame of the video. Initialization essentially involves finding the parameters for the cylinder (the radius and the height). In the current impelementation, we assume that the face is roughly frontol during initialization. We use the optimal edge-based shape detection algorithm [17] to detect the face in the first frame. This algorithm looks for ellipses containing facial features for face detection using the optimal shape operator.

3 Features

The choice of features is extremely important for the task of 3D pose estimation of a moving face, probably second only to that of the structural model. More than anything else, the features should be easy to detect. In addition, ideally they should be robust to occlusions, and changes in pose, expression and illumination. Humans detect and track faces (known or unknown) effortlessly using features like eyes, nose, mouth, hair etc. For machines, this might not be easy. In a monocular video, only input the machines have is an image which is a 2D projection of the current appearance of the face. The appearance of the features used by humans changes a lot with variations in pose, expression etc. In fact sometimes, few of the features are not even visible in the image. This makes the automatic detection and thereby tracking of these features very difficult.

As stated previosuly, ours is a hybrid approach which tries to make use of the advatanges of a purely geometric approach (useful when partial/complete information about the geometric structure of the object is available) and that of stastical inference. In this work, we stress-test this approach using an extremely simple and easily computable feature. We superimpose a rectangular grid all around the curved surface of our elliptical cylinder. Then mean intensity is computed for each of the visible grids which forms the feature vector. Note that many of the mean values will be undefined which correspond to the the part of the face which is not visible in the frame.

Though quite simple, the feature vector is not all that bad when viewed from the point of view of our framework. First of all, it is easily computable. Given the current configuration of the face, the grids can be projected onto the image frame and the mean can be computed for each of them. This might seem suspect as the current configuration of the face is not available! Rather that is what we are trying to estimate. Crudely speaking, we first predict the current configuration based on the past configuration and then test its likelihood using the current feature vector. This will become clear once we present the particle-filter framework where each particle represents a configuration of the face. The *mean vector*, by itself, is not invariant to pose but pose is not an issue in our framework as long as the cylindrical assumption is fine. Mean is definitely not invariant to illumination changes. We use robust statistics to make the approach robust to illumination. The fact that the mean is computed for lots of small regions makes it an appropriate substrate for robust statistics. The basic idea here is that illumination does not affect the algorithm as long as many of the *means* remain unaffected. The same idea works even for handling partial occlusions and expression changes.

4 Tracking Framework

Once the structural model and feature vector have been fixed, the goal is to estimate the configuration (or pose) of the moving face in each frame of a given video. Breaking this down to each frame, one can see that only information available to perform the desired estimation is the face configurations in the previous frames and the current observation (the current frame). This can be viewed as a dynamic state estimation problem. Here the state consists of the six configuration parameters: three for the translation and three for the orientation of the face. The Bayesian approach to handle this problem is to gather the available information to come up with the probability density function (pdf) of the state. This estimation can be done recursively for each frame using particle filters.

4.1 Particle Filter

Particle filtering [6][8] is an inference technique for estimating the unknown dynamic state θ of a system from a collection of noisy observations $y_{1:t}$. Quite often, a state space model is used to perform this estimation. The two components of this approach are the state transition model which models the state evolution, and the observation model which specifies the state-observation dependence:

$$\text{State transition model: } \theta_t = f(\theta_{t-1}, u_t), \qquad (1)$$

$$\text{Observation model: } y_t = g(\theta_t, v_t), \tag{2}$$

where u_t is the system noise while v_t is the observation noise. In general, the functions f and g can also be time-dependent. The particle filter approximates the desired posterior pdf $p(\theta_t|y_{1:t})$ by a set of weighted particles $\{\theta_t^{(j)}, w_t^{(j)}\}_{j=1}^N$, where N denotes the number of particles. The state estimate $\hat{\theta}_t$ can be recovered from the pdf as the maximum likelihood (ML) estimate or the minimum mean squared error (MMSE) estimate or any other suitable estimate based on the pdf.

To keep the tracker as generic as possible, we use a random-walk model as the motion model:

$$\theta_t = \theta_{t-1} + u_t, \tag{3}$$

where u_t is normally distributed about zero. Based on the domain knowledge, one can come up with a motion model that will be capable of estimating the pdf better with lesser number of particles. For example, if the task is to track the face of a spectator in a tennis match, a motion model heavily biased towards *yaw* might be a better choice than a generic model.

The observation model involves the feature vector described in the previous section. In our framework, we can rewrite the observation equation as:

$$z_t = \Gamma\{y_t; \theta_t\} = F_t + v_t, \tag{4}$$

where y_t is the current frame (the grayscale image), Γ is the mapping that computes the feature vector given an image y_t and a configuration θ_t, z_t is the computed feature vector and F_t is the feature model. The feature model is used to compute the likelihood of the particles (which correspond to different proposed configurations of the face). For each particle the likelihood is computed using the average sum of square differences (SSD) between the feature model and the *mean vector* z_t corresponding to the particle.

On one extreme, the feature model can be a fixed template (say, the feature vector corresponding to the first frame i.e., $F_t = F_0$) while on the other hand one can use a dynamic template e.g, the feature vector belonging to the best particle at the previous frame i.e., $F_t = \hat{z}_{t-1}$. Similar to [11], we refer to the fixed template $F_t = F_0$ as the lost model while the dynamic component $F_t = \hat{z}_{t-1}$ as the wander model. It is worthwhile to note that though the lost component should be credible (assuming initialization is good), quite often it is not capable to handle the appearance changes due to illumination, expression, etc. as the face translates/rotates in the real world. On the other hand, the dynamic nature of the wander component makes it suitable to take care of appearance changes but it is susceptible to drifts. This means that if we have a bad estimation for a frame, it becomes very diffcult for the tracker to correct itself in subsequent frames. We use a combination of both which provides resiliency to our tracker. Resiliency is a very important property of any tracker as however good a tracker is, it can always loose track due to an unexpected change in conditions. In the current implementation, the likehood of a particle is computed as the maximum of the likehoods using the lost model and the wander model. The prior is biased towards the lost model by $0.52 : 0.48$. We take the maximum of the two likelihoods instead of mixing the two to avoid boosting up the probability of a bad particle accidentally. This gives us the capability both to handle appearance changes and to correct the estimation even if the wander model drifts. The number of particles used were typically in the range 200-500.

Before moving further, we need to revisit the issue regarding the knowledge of focal length which was left unanswered earlier. Intuitively speaking, if for the current frame we have particles predicted with approximately correct translation values (with the assumed arbitrary focal length), the orientation estimation should not be wrong. This comes from the observation that for the particles with correct translation, the face essentially undergoes only rotation. Note that the translation values are recovered only upto a scale factor but one cannot do better as far as translation is concerned even with the knowledge of focal length. Experiments show that the results obtained are impressive even with the arbitrary (but suitable) focal length.

4.2 Robust Statistics

The performance of the filtering method described is limited by the appropriateness of the likelihood model. If the feature vector or the method of likelihood computation is not good enough to distinguish between different configurations of the face, tracking can not be expected to be good. Furthermore, we use the mean of grids as the feature vector which by itself is not robust to occlusions, illumination, expression etc. The fact that lots of means are computed over small local regions makes the scenario suitable for the application of robust statistics in the likelihood computation. In the current implementation, we trust only the top half of the means and treat the rest as outliers. The robustified likelihood computation can be represented as:

$$p(y_t|\theta_t^{(j)}) = e^{-\lambda dist} \tag{5}$$

$$\text{where,} \quad dist = \frac{\sum_{m,n} \eta(m,n) d(m,n)}{\sum_{m,n} \eta(m,n)} \tag{6}$$

where $\eta(m,n)$ is 1 if the $(m,n)^{th}$ grid is visible in both the model and the particle and 0 otherwise, while $d(m,n)$ is computed as:

$$d(m,n) = \begin{cases} (F_t(m,n) - z_t^{(j)}(m,n))^2 & \text{if } d(m,n) < c \\ c & \text{otherwise} \end{cases}$$

$$\text{where,} \quad c = median(\{d(m,n)\}) \tag{7}$$

5 Experiments and Results

We conducted three different experiments to show the efficacy of our tracking approach. The experiments are designed to display the ability of the tracker to handle occlusion, expressions and extreme poses. The comparison with the ground truth is also done. In addition, we show how maintaining 3D correspondences help in other problems like recognition.

5.1 Tracking under extreme poses

We conducted tracking experiments on 3 datasets (Honda/UCSD dataset [14], BU dataset [3] and Li dataset [15]). These datasets have numerous sequences in which there are significant illumination changes, expression variation and people are free to move their heads arbitarily. Figure 2 shows few of the frames from several videos with grid points on the estimated cylinder overlaid on the image frame. The first row shows the ability of the tracker to accurately estimate the pose even under extreme poses. Most 2D approaches would not be able to maintain track under such severe poses. The second row shows some frames in which the the tracker was able to maintain track inspite of severe occlusion. The subject waved his hand across his face while simultaneously turning his head. The robust statistics employed in the likelihood computation enables the tracker to maintain track under occlusion. Moderate expressions do not affect our feature since it is the mean intensity within a small surface patch on the face. During certain severe expression changes robust statistics helps us maintain the track. The third and fourth rows show more tracking results from the BU dataset and the Li dataset, respectively. In the fourth row, the subject is removing his glasses while rotating his head. Our tracker is able to maintain the track all along the sequence.

5.2 Ground Truth Comparison

The BU dataset [3] provides us with ground truth values for the pose of the face in each frame. We conducted tracking experiments on the BU dataset and compared yaw, pitch and roll estimated by our tracker to the ground truth. Figure 3 shows the comparison between the estimated pose of the face and ground truth for six different sequences in the dataset. We see that the tracker accurately estimates the pose of the face in most of these frames.

5.3 Recognition with non-overlapping poses

Most methods for recognition require that the gallery contains an instance of a face with a pose very similar to the one in the probe. Since our tracking method maintains explicit pose of the face during each frame, we do not need to have the same poses seen in the gallery and the probe. In this experiment we show this by performing recognition on non-overlapping poses. The gallery consists of a video sequence of about 10-15 frames in which the individual turns his head left from about 15 degrees away from from frontal to extreme left. The probe consists of a video sequence in which the individual turns his head right from 15 degrees away from frontal all the way to the right. Therefore, there is no pose overlap between the gallery and the probe. In fact, the closet poses in the gallery and the probe differ by atleast 30 degrees. We used 10 subjects from the Honda/UCSD dataset [14] for this experiment. For each frame we build a texture mapped cylinder using the tracked pose. We used the minimum sum of squared distance between a gallery model and a probe model as the distance between two videos. This is a very challenging experiment since the poses exhibited by the gallery videos and those exhibited by the probe videos are very different. Therefore, the similarity matrix obtained in this experiment was weakly diagonal. Inspite of this, we obtained 100% recognition rate in this experiment, i.e., all the 10 probe videos were recognized correctly. This is very promising and we hope to extend the results to a larger dataset for arbitary uncontrolled videos of individuals.

6. Summary and Conclusions

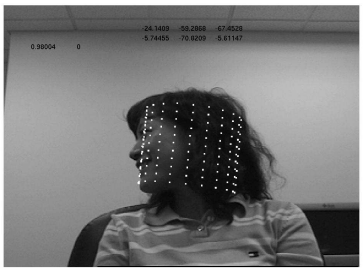
In this paper, we have proposed a method for tracking the facial pose in a video. The tracker is robust to occlusions and illumination changes and maintains track even during extreme poses. We have also shown, how such 3D pose tracking can help in problems like face recognition from videos. We are currently working on using this as a front-end to applications like face recognition and facial expression analysis.

References

[1] anonymous. anonymous. *anonymous*, anonymous:anonymous, anonymous.

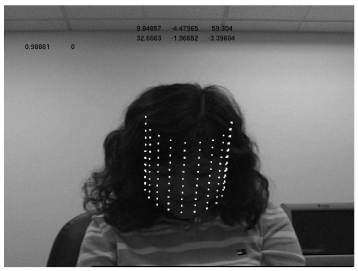
[2] S. Birchfield. An elliptical head tracker. In *Proceedings of the 31st Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, California*, pages 1710–1714, November 1997.

[3] M. L. Cascia, S. Sclaroff, and V. Athitsos. Fast, reliable head tracking under varying illumination: An

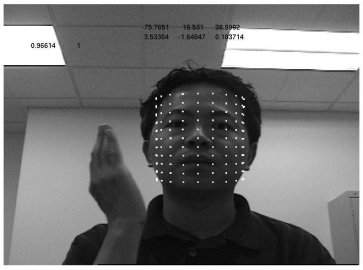


Frame # 73 − (−5, −70, −5)

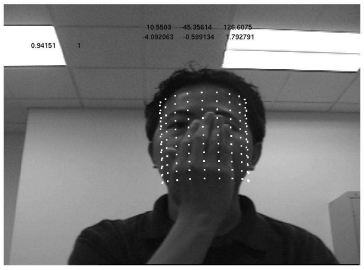
Frame # 93 − (5, 9, −37)



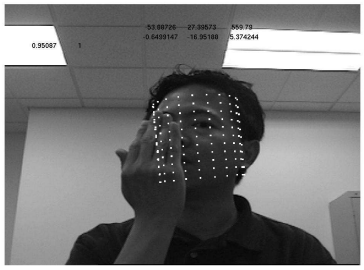
Frame # 127 − (−3, −2, 32)



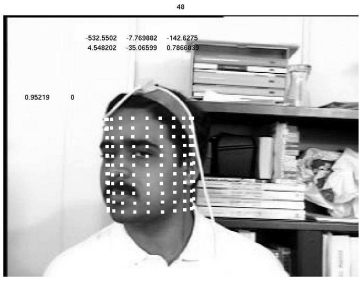
Frame # 20 − (0, −1, 3)



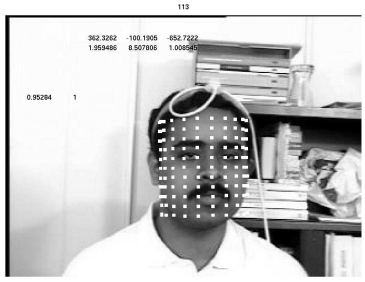
Frame # 32 − (2, 0, −4)



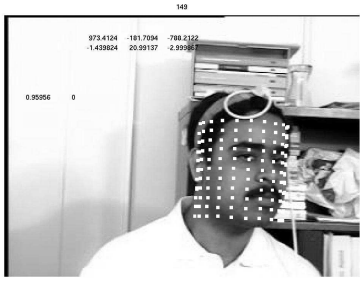
Frame # 114 − (5, −17, −1)



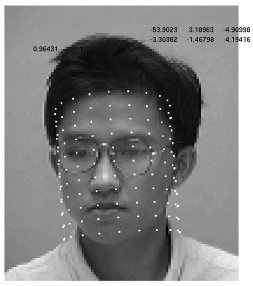
Frame # 48 − (1, −35, 4)



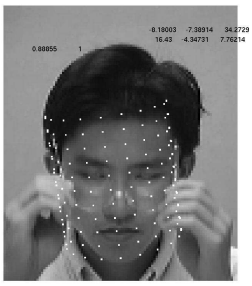
Frame # 113 − (1, 8, 2)



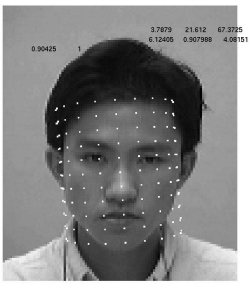
Frame # 149 − (−3, 21, −1)



Frame # 433 − (4, −2, −3)



Frame # 445 − (7, −4, 16)



Frame # 448 − (4, 1, 6)

Figure 2: Tracking results on different datasets under severe occlusion, extreme poses and different illumination conditions. The cylindrical grid is overlaid on the image plane to display the results. Each frame is labeled with its frame number in the video. The 3-tuple shows the estimated orientation (roll, yaw, pitch) in degrees for each of the frames.

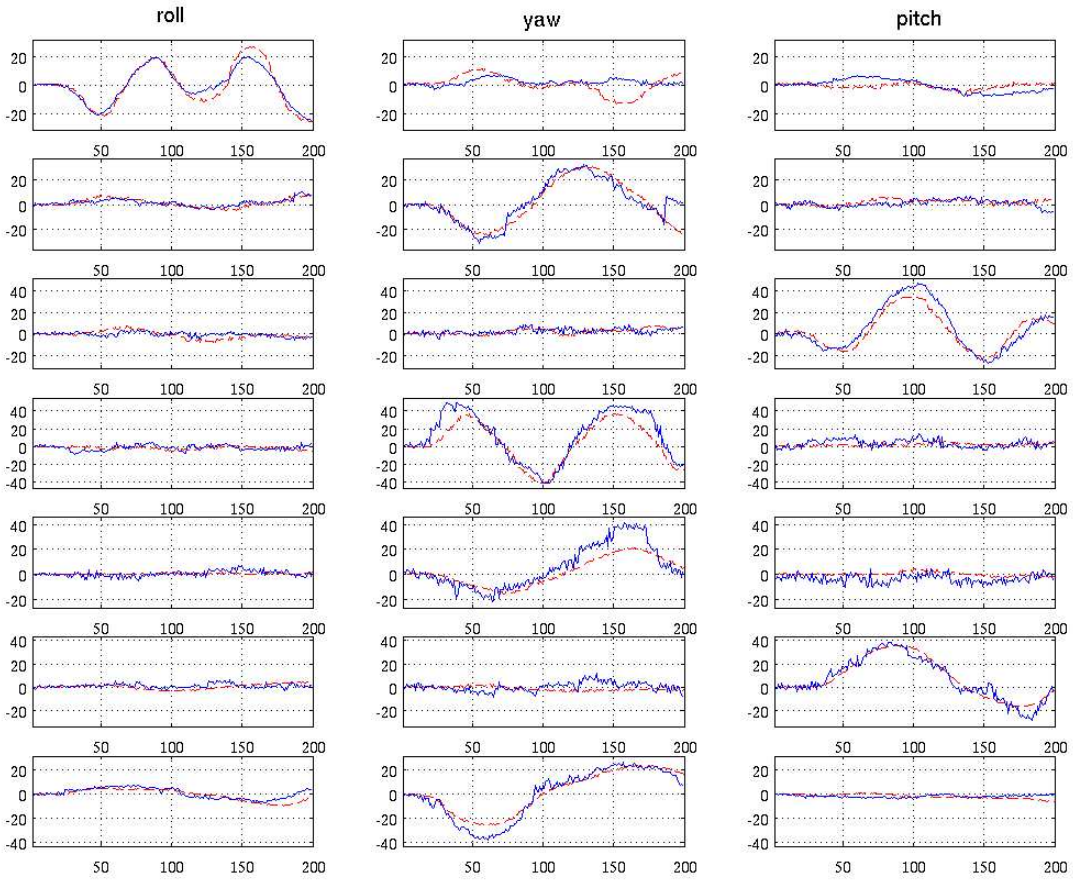


Figure 3: Comparison with the ground truth. Each row corresponds to one video displaying the three orientation parameters. The red/dashed curve depicts the ground truth while the blue/solid curve depicts the estimated values.

approach based on registration of texture-mapped 3D models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(4):322–336, April 2000.

[4] J. L. Crowley and F. Berard. Multi-modal tracking of faces for video communication. In *IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR*, 1997.

[5] F. Dornaika and J. Ahlberg. Fast and reliable active appearance model search for 3D face tracking. *IEEE Transactions on Systems, Man and Cybernetics–Part B: Cybernetics*, 34(4):1838–1853, August 2004.

[6] A. Doucet, N. D. Freitas, and N. Gordon. *Sequential Monte Carlo methods in practice*. Springer-Verlag, New York, 2001.

[7] P. Fieguth and D. Terzopoulos. Color-based tracking of heads and other mobile objects at video frame rates. In *IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR*, 1997.

[8] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings on Radar and Signal Processing*, volume 140, pages 107–113, 1993.

[9] G. D. Hager and P. N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(10):1025–1039, October 1998.

[10] T. S. Jebara and A. Pentland. Parameterized structure from motion for 3D adaptive feedback tracking of faces. In *IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR*, 1997.

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

[12] A. Lanitis, C. Taylor, and T. Cootes. A unified approach for coding and interpreting face images. In *International Conference on Computer Vision, Cambridge, MA*, pages 368–373, 1995.

[13] A. Lanitis, C. Taylor, and T. Cootes. Automatic interpretation and coding of face images using flexible models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):743–756, July 1997.

[14] K. C. Lee, J. Ho, M. H. Yang, and D. Kriegman. Video-based face recognition using probabilistic appearance manifolds. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[15] B. Li and R. Chellappa. Face verification through tracking facial features. *Journal of the Optical Society of America A*, 18:2969–2981, December 2001.

[16] L. Lu, X. Dai, and G. Hager. A particle filter without dynamics for robust 3D face tracking. In *IEEE Conference on Computer Vision and Pattern Recognition, Washington, D.C.*, 2004.

[17] H. Moon, R. Chellappa, and A. Rosenfeld. Optimal edge-based shape detection. *IEEE Trans. on Image Processing*, 11(11):1209–1226, 2002.

[18] Y. Moses, D. Reynard, and A. Blake. Robust real time tracking and classification of facial expressions. In *International Conference on Computer Vision, Cambridge, MA*, pages 296–301, 1995.

[19] N. Oliver, A. Pentland, and F. Berard. Lafter: Lips and face real time tracker. In *IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR*, 1997.

[20] F. Pighin, R. Szeliski, and H. Salesin. Resynthesizing facial animation through 3D model-based tracking. In *Seventh International Conference on Computer Vision, Kerkyra, Greece*, pages 143–150, 1999.

[21] A. L. Yuille, D. S. Cohen, and P. W. Hallinan. Feature extraction from faces using deformable templates. In *International Conference on Pattern Recognition*, 1994.

[22] S. Zhou, V. Krueger, and R. Chellappa. Probabilistic recognition of human faces from video. *Computer Vision and Image Understanding (CVIU) (special issue on Face Recognition)*, 91:214–245, 2003.