3D Head Pose Estimation with Symmetry based Illumination Model in Low Resolution Video

Martin Gruendig[1] and Olaf Hellwich[2]

[1] Robert Bosch GmbH, FV/SLH, P.O. Box 777 777, 31132 Hildesheim, Germany
[2] Computer Vision and Remote Sensing TU Berlin, 10623 Berlin, Germany

Abstract. A head pose estimation system is described, which uses low resolution video sequences to determine the orientation and position of a head with respect to a internally calibrated camera. The system employs a feature based approach to roughly estimate the head pose and an approach using a symmetry based illumination model to refine the head pose independent of the users albedo and illumination influences.

1 Introduction

3D head pose estimation and tracking from monocular video sequences is a very active field of research in computer vision. In this paper we want to introduce a 3D head pose estimation system which is designed to initialize a tracking framework to track arbitrary movements of a head. This paper concentrates on the initialization part of our system which has to work on low resolution video sequences. The head usually covers 60x40 pixels in the images, and it has to be robust with respect changes in illumination, facial gestures and different users. A number of different approaches have been proposed for the problem of 3D head pose estimation and tracking. Some using a 3D head model and tracking distinct image features through the sequence [2], [7], [4], [1]. The image features correspond to anchor points on the 3D head model which then can be aligned accordingly and the head pose is estimated. Another approach is to model 3D head movement as a linear combination of a set of bases, that are generated by changing the pose of the head and computing a difference image of the poses [14]. The coefficients of the linear combination that models the difference image best is used to determine the current head pose. A third popular approach is to employ optical flow constrained by the geometric structure of the head model [6], [5], [16]. Since optical flow is very sensitive with respect to illumination changes, [10] also included an illumination basis to model illumination influences in his optical flow approach. Except for [6] all approaches work on high resolution images. Except for [2] none of the mentioned approaches includes an automatic initialization of the tracking without the user to keep still and to look straight into the camera.

There are a number of 3D face pose estimation approaches. Systems which do not require high resolution images of the face, either lack the required accuracy which is needed to initialize a tracking system [8], or are not illumination and person invariant [13].

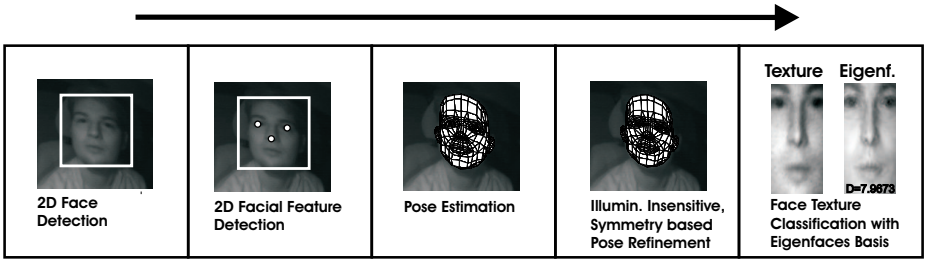

Fig. 1. Structure of head pose estimation system.

2 Motivation

A fully automatic head pose tracking system has to automatically initialize the tracking. This includes a reliable estimation of the 3D head pose without any customized model information. Some systems require the user to look straight into the camera to initialize [7], [14], [16], others only determine the relative change of pose with respect to the pose in the first frame [4], or the user is required to identify certain features like eyes and nose manually [5]. Since we intend to automatically initialize and track a head, in low resolution video sequences, all the above approaches are not an option. Our goal is to reliably estimate the head pose with respect to the camera if the deviation of the current orientation from a frontal view does not exceed 30 degrees. In case these conditions are not met by the current head pose, no initialization should occur until we reach a frame with a head pose that does meet the condition. One can think of it as a trap. The initialization process is divided into five parts, see Fig. 1

3 Implemented System

3.1 2D Face Detection

First we employ a face detection algorithm that is capable of detecting faces that meet our orientation condition. We use the OpenCV implementation [$http://sourceforge.net/projects/opencvlibrary$] of the detection algorithm proposed by Viola and Jones [11] which works very fast and reliable. The face detection gives us a region of interest (ROI) in the image which is passed on to the next step of the initialization.

3.2 Facial Feature Detection and Rough Pose Estimation

In order to roughly estimate the current head pose we intend to detect the image positions of the eyes and the tip of the nose. A radial symmetry interest operator is employed [3] on the upper part of the ROI to detect possible eyes. Since in the low resolution images, each eye is usually a dark radial feature surrounded

by a brighter area, eyes yield a large answer in the radial symmetry analysis. It is still difficult though to make an exact prediction for the eyes. Instead of taking the two largest local maximums from the radial symmetry analysis we rather allow 15 hypotheses of possible eye positions at this stage, to be sure to include the correct ones. The same strategy is used for the tip of the nose. Due to its prominent position, the tip of the nose reflects light very well and usually appears as a bright radial feature on a darker background. 3 hypotheses usually suffice for the nose to include the correct position.

For every combination of 2 eyes and a nose we can compute a resulting head pose using a weak geometry projection. We have the 3D object coordinates of the eyes and the nose on a 3D head model of an average person, and the 3 corresponding image positions of the combination. Having computed the pose of every combination, we can discard all the combinations which deviate more than 30 degrees from a frontal view. These heuristics usually do reduce the number of relevant combinations significantly. The remaining combinations are evaluated. For this evaluation we use a database of 3000 different eyes and noses in gabor wavelet space [15]. Each eye hypothesis and nose hypothesis is compared against the database and receives a final feature score. This feature score is the similarity value of the database entry that fits best. The sum of the feature-scores of the combination yields the combination-score. The associated pose of the combination that received the highest combination-score is an estimate of the current head pose.

3.3 Symmetry Considerations

For the refinement of the initialization a novel, illumination and albedo insensitive, symmetry based approach is employed. First we assume that the texture of the right half of the face is symmetric to the left. By projecting a 3D model of a face under the estimated pose into the image, we can extract the underlying texture of the face from the image. Now consider a point p on a lambertian surface. Ignoring attatched shadows, the irradiance value E_p of the surface point p is given by

$$E_p = k_p \left(\boldsymbol{N}_p \cdot \boldsymbol{L}_p \right) \tag{1}$$

where k_p is the nonnegative absorption coefficient (albedo) of the surface at point p, $\boldsymbol{N}_p$ is the surface normal at point p, and $\boldsymbol{L} \in \mathbb{R}^3$ characterizes the collimated light source, where $\|\boldsymbol{L}\|$ gives the intensity of the light source. The gray level intensity I_p measured by a camera is an observation of E_p. We can therefore write

$$I_p \sim k_p \left(\boldsymbol{N}_p \cdot \boldsymbol{L}_p \right) \tag{2}$$

We now assume that a face is symmetric with respect to a mirror axis along the nose. Therefore we can assume

$$k_{pr} = k_{pl} = \frac{I_{pr}}{(\boldsymbol{N}_{pr} \cdot \boldsymbol{L})} = \frac{I_{pl}}{(\boldsymbol{N}_{pl} \cdot \boldsymbol{L})} \Rightarrow \frac{I_{pr}}{I_{pl}} = \frac{(\boldsymbol{N}_{pr} \cdot \boldsymbol{L})}{(\boldsymbol{N}_{pl} \cdot \boldsymbol{L})} \tag{3}$$

where k_{pr} is the albedo of a point pr on the right side of the face and k_{pl} is the albedo of the symmetrically corresponding point pl on the left side of the face. Fig. 2 illustrates the computation.

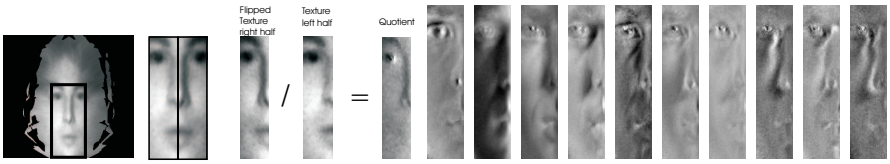

Fig. 2. *From left to right:* Extracted texture of the head under the current pose. Face texture divided into right and left half. Division of flipped right half of the face texture and the left half of the face texture. Quotient. *Right:* Illumination basis with 10 basis vectors that approximately span the space of albedo independent face texture quotients

3.4 Illumination Model

Following the symmetry considerations, we can now generate a parametric illumination model for human faces. The following is related to [14] and [10]. In contrast to [14] and [10] we do not generate the parametric illumination model based on the textures themselves. In order to achieve user independence we use the fraction $H(I) = \frac{I_r}{I_l}$. If we extract the face texture in form of a vector I_j from a set of images of a face illuminated from a different direction in each image j. We can then compute the fraction

$$H(I) = \frac{I_r}{I_l} \tag{4}$$

for each element of these textures $\boldsymbol{I}_j$. By performing a singular value decomposition on the texture fractions $\boldsymbol{H}_j$ we can generate a small set of 10 basis vectors b to form a illumination basis, so that every fraction $\boldsymbol{H}_j$ can be expressed as a linear combination of the columns of $\boldsymbol{B}$

$$\boldsymbol{B} = [\boldsymbol{b}_1|\boldsymbol{b}_2|\ldots|\boldsymbol{b}_{10}] \tag{5}$$

$$\boldsymbol{H}_j = \boldsymbol{B}\boldsymbol{w} \tag{6}$$

where $\boldsymbol{w}$ is the vector of linear coefficients. Fig. 2 illustrates the illumination basis. Note that the fraction $H = \frac{I_r}{I_l}$ is a value that is independent of the persons individual reflectance parameters (albedo). Therefore we do not need to know the persons albedo, in that aspect this new approach differs from [12] and [10]. Using this measure, it is therefore possible to get a albedo independent model of illumination influences of a face. In contrast to [14] where a normally illuminated face of the current person is subtracted from each illumination texture in order

to build a user independent illumination basis, we strictly model illumination with the lambertian illumination model without the need to assume an additive nature of illumination influences. 10 basis vectors seem to suffice to account for the non lambertian influences and self shadowing.

3.5 Head Pose Refinement

In order to further refine the pose estimation we can now use the user independent illumination model and formulate the pose refinement as a least squares problem in the following way. Let $\boldsymbol{X}_i \in \mathbb{R}^3$ be a point on the head in 3D space with respect to a head centered model coordinate system. $\boldsymbol{x}_i \in \mathbb{R}^2$ is the corresponding 2D point in the image coordinate system.

$\boldsymbol{X}_i$ is projected onto $\boldsymbol{x}_i$ under the current head pose μ, which consists of the orientation $[\mu_1; \mu_2; \mu_3]$ and the 3D position $[\mu_4; \mu_5; \mu_6]$ of the head with respect to the camera.

$$\tilde{\boldsymbol{X}}(\boldsymbol{X}, \mu) = R(\mu_1, \mu_2, \mu_3)\boldsymbol{X} + t(\mu_4, \mu_5, \mu_6) \tag{7}$$

The similarity transform in (7) aligns the model coordinate system with respect to the camera where R is a rotation about the angles μ_1, μ_2, μ_3 and t is the translation μ_4, μ_5, μ_6 that translates the origin of the model coordinate system to the origin of the camera coordinate system. The collinearity equation

$$\boldsymbol{x}(\tilde{\boldsymbol{X}}) = \left[\frac{k_1^T \cdot \tilde{\boldsymbol{X}}}{k_3^T \cdot \tilde{\boldsymbol{X}}} \, , \, \frac{k_2^T \cdot \tilde{\boldsymbol{X}}}{k_3^T \cdot \tilde{\boldsymbol{X}}} \right]^T ; \; K = \begin{bmatrix} k_1^T \\ k_2^T \\ k_3^T \end{bmatrix} \tag{8}$$

formulates the central projection of the aligned model coordinates $\boldsymbol{X}_i$ into the image coordinates $\boldsymbol{x}_i$, where K is a matrix containing the intrinsic camera calibration parameters. The gray value intensities I of the image can be formulated as a function of $\boldsymbol{x}$ (9). H can therefore be expressed as a function of μ (10).

$$I(\boldsymbol{x}) = I \tag{9}$$

$$H(\mu) = H(I(\boldsymbol{x}(\tilde{\boldsymbol{X}}(\mu)))) \tag{10}$$

We can now formulate the objective function O that needs to be minimized. We want to find a head pose μ that can be confirmed by the illumination basis B as well as the roughly detected feature positions of the eyes and the nose in the image $\boldsymbol{x}_n$ in a least squares sense. We therefore set

$$O(\mu, w) = \sum_n P_n(\boldsymbol{x}_n(\mu) - \boldsymbol{x}_n)^2 + \sum_i P_i(H_i(\mu) - B_i(w))^2 \tag{11}$$

P_i and P_n are weights associated with every element i of the illumination basis and every feature point n of the detected feature positions. With these weights

we can control how much influence the feature points have with respect to the illumination basis in the least squares adjustment. To equally weigh the feature points and the illumination basis and since we have 3 detected feature points we usually set $P_i = 1$ and $P_n = \frac{1}{3}\sum P_i$.

Since O is nonlinear, we need to expand the function into a taylor series in order to be able to iteratively solve the least squares adjustment in the fashion of a Gauss-Newton Optimization. As a starting point we can use the pose estimation from the feature based head pose estimation μ_0.

By setting

$$A = \begin{bmatrix} \nabla H & -\nabla B \\ \nabla \boldsymbol{x} & 0 \end{bmatrix}; \delta l = \begin{bmatrix} H(\mu_0) - B(w_0) \\ \boldsymbol{x}(\mu_0) - \boldsymbol{x} \end{bmatrix}; P = \begin{bmatrix} diag(P_i) & 0 \\ 0 & diag(P_n) \end{bmatrix} \quad (12)$$

where A is a Matrix that includes the jacobians, δl is a vector and P is the diagonal matrix with the associated weights.

$$O(\delta\mu, \delta w) = \left\| P(A\left[\delta\mu, \delta w\right]^T + \delta l) \right\| \quad (13)$$

$$\left[\delta\mu, \delta w\right]^T = -(A^T P A)^{-1} A^T P \delta l \quad (14)$$

$$\mu_0 = \delta\mu + \mu_0 \;\; ; \;\; w_0 = \delta w + w_0 \quad (15)$$

Equation (13) formulates the linearized objective function in matrix notation. Solving the set of equations $\nabla O = 0$ gives the solution in (14). Equation (15) yields the update of the head pose μ and the illumination basis coefficients w for the next iteration step. Usually 10 iterations suffice for the adjustment to converge. After each iteration step a visibility analysis is performed to determine for which points on the face both symmetrically corresponding points on the left half and on the right half are visible under the current pose μ. If either of the two symmetrically corresponding points is not visible, the pair of points is excluded for the next iteration step. This way we can handle self occlusion.

3.6 Face Texture Verification

In order to evaluate the detected pose of the head and to discard gross orientation errors it is crucial to verify the detected pose. In our approach we use the face texture which was extracted from the image under the current head pose as a verification hint. By calculating the distance D from an eigenface basis of face textures [9], we can evaluate the current face texture. At this stage we assume that a correct head pose yields a face texture with a very low distance from the eigenface basis, hence these two measures are correlated. The eigenface basis was constructed from a database with 3000 high resolution images of 20 different people under varying lighting conditions and facial expressions.

After the head pose refinement the distance D of the current face texture from the eigenface basis can be used to classify the estimated head pose as correct or as incorrect. A key feature of the system design is therefore the threshold T_D for the classification $h(D)$

$$h(D) = \begin{cases} 1 & D < T_D \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

If we set T_D to a very high value, we will get a relatively large number of false positives. If we set T_D to a very low value we will get a relatively small number of false positives but the fraction of true negatives will increase, which leads to a lower detection rate. Since the application we have in mind is to initialize a head pose tracking system, we are not explicitly interested in a very high detection rate. If the initialization fails in one frame, it is possible to simply try again in the next frame. The overall detection rate will significantly increase if several independent frames are evaluated instead of only one.

4 Experiments and Results

So far the system only works off line as a prototype in a Matlab implementation. We are confident though to achieve real time performance in a C implementation. The most time consuming part is the least squares head pose refinement, since it is a non linear adjustment with an iterative solution. Similar implementations of least squares approaches have achieved real time performance before though [6].

To test our system in real world conditions we recorded several sequences of 12 different people in low resolution, Fig. 3. The head usually covers 60x40 pixel. These images include all possible variations of face appearance. Arbitrary poses, facial expressions, different illumination conditions, and partial occlusions of the face are sampled by these images. In order to generate a ground truth, we manually determined the head pose in $j = 1500$ of these images by labeling the center of the eyes, the tip of the nose and the corners of the mouth and calculating the six parameters $[\bar{\mu}_1; \bar{\mu}_2; \bar{\mu}_3; \bar{\mu}_4; \bar{\mu}_5; \bar{\mu}_6]$ of the ground truth pose $\bar{\mu}$ from that. The mean errors of this ground truth is given in table 1. Table 1 also lists the mean errors of the rough pose estimate, the refined pose estimate and the refined pose estimate with a texture verification threshold set to $T_D = 11$. The mean errors decrease with each step of the system. It is also worth mentioning that the mean errors of our system with respect to the ground truth correspond to the accuracies of the ground truth itself, table 1. In other words, the real accuracies of our system might even be better. Fig. 3 shows the mean errors of the rotational and the translational pose parameters. The diagrams indicate a decreasing accuracy of the pose if the threshold T_D is set to high values in order to increase the detection rate and therefore the robustness. We can increase the robustness by performing the procedure on several subsequent frames and only taking the frame into account which received the lowest value in the distance from the eigenface basis D. This increases the detection rate and decreases the false positive rate. Fig. 3 shows a ROC diagram for setups with 1 frame, 2 frames and 6 frames. Fig. 3 also shows 3 samples of the test results.

5 Conclusion

We introduced a system to estimate the head pose in 6 degrees of freedom in low resolution images. The system is designed to automatically initialize a 3D head

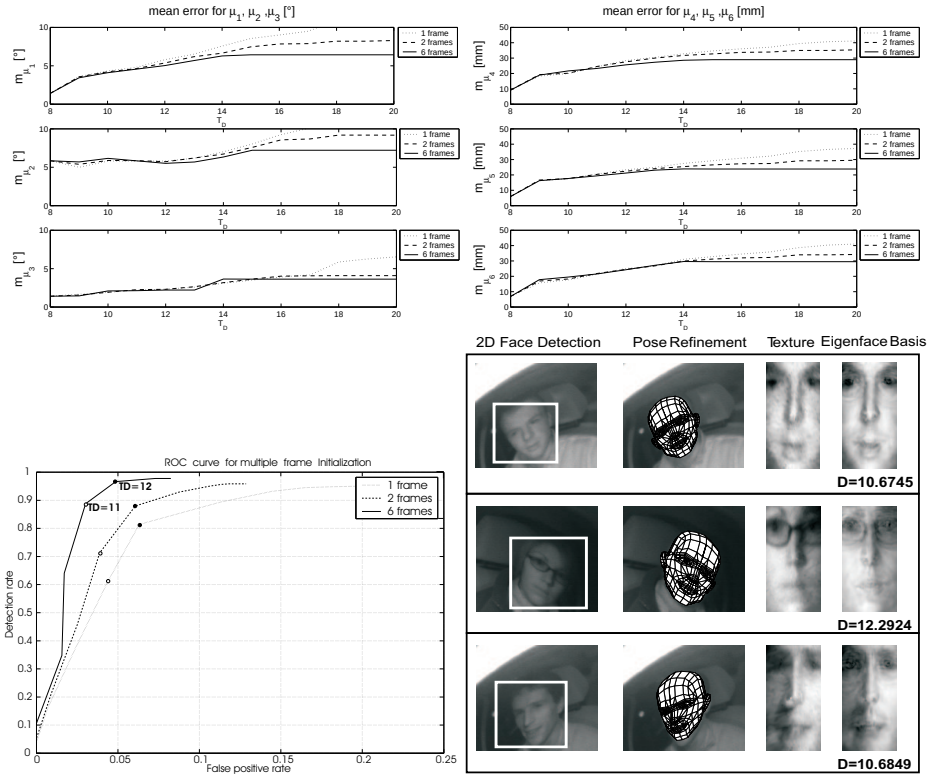

Fig. 3. *Top left and right:* Mean errors of parameters of the pose μ, with respect to the threshold T_D for setups with 1 frame, 2 frames and 6 frames. As the diagrams indicate, the accuracy of the system can not be improved by taking more frames into account. *Bottom left:* ROC diagram for setups with 1 frame, 2 frames and 6 frames. 2 discrete values for the thresholds T_D of the face Texture classification are plotted as a white dot for the value $T_D = 11$ and as a black dot for $T_D = 12$. The best results were achieved for a setup with 6 frames and a threshold $T_D = 11$. With fewer frames taken into account, the results gently decrease in quality. *Bottom right:* 3 samples of the test results. The results of the face detection the pose refinement and the face texture verification are displayed.

Table 1. Mean Errors

Mean Errors	m_{μ_1}	m_{μ_2}	m_{μ_3}	m_{μ_4}	m_{μ_5}	m_{μ_6}
Ground truth	6 deg.	6 deg.	3 deg.	27 mm	26 mm	25 mm
Rough Head Pose Estimate	10 deg.	13 deg.	5 deg.	38 mm	41 mm	40 mm
Refined Head Pose	8 deg.	7 deg.	3 deg.	30 mm	28 mm	29 mm
Refined with Texture verificat. $T_D = 11$	6 deg.	6 deg.	3 deg.	28 mm	22 mm	24 mm

pose tracking system, e.g. as in [6]. The system is independent of illumination influences and requires no personalization training or user interaction. Since the system is based on global face symmetry only head poses in which both eyes are visible will be detected. In our experiments on low resolution images, we achieved a detection rate of 90% at a false positive rate of 3% if 6 subsequent frames are taken into account. Considering only one single frame we achieved a detection rate of 70% at a false positive rate of 6%. Our experiments indicated mean orientation errors of $m_{\mu_1} = m_{\mu_2} = 6$ degrees and of $m_{\mu_3} = 3$ degrees respectively. The mean positioning errors are about 25 mm in each dimension. This matches at least the accuracy of manual head pose estimation in low resolution images.

References

1. R. Cipolla A. Gee. Fast visual tracking by temporal consensus. *Image and Vision Computing*, 14:105–114, 1995.
2. T. Jebara A. Pentland. Parameterized structure from motion for 3d adaptive feedback tracking of faces. Technical Report,Media Laboratory, MIT, 1996.
3. G. Loy A. Zelinsky. Fast radial symmetry for detecting points of interest. *IEEE Transactions on Pattern analysis and Machine Intelligence*, 25(8):959–973, 2003.
4. Natsuko Matsuda Charles S. Wiles, Atsuto Maki. Hyperpatches for 3d model acquisition and tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12):1391–1403, 2001.
5. D. Metaxas D. DeCarlo. The integration of optical flow and deformable models with applications to human face shape and motion estimation. IEEE Conference on Computer Vision and Pattern Recognition, 1996.
6. T. Kanade J. F. Cohn, J. Xiao. Robust full motion recovery of head by dynamic templates and re-registration techniques. Automated Face and Gesture Recognition, 2002.
7. S. Basu A. Pentland J. Stroem, T. Jebara. Real time tracking and modelling of faces: An ekf-based analysis by synthesis approach. Proceedings of the Modelling People Workshop at ICCV'99, 1999.
8. Y. Wu K. Toyama. Wide-range, person and illumination insensitive head orientation estimation. Automated Face and Gesture Recognition, 2000.
9. A. P. Pentland M. A. Turk. Face recognition using eigenfaces. CVPR92, 1991.
10. G. D. Hager P. N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Transactions on Pattern analysis and Machine Intelligence*, 20(10):1025–1039, 1998.
11. M. Jones P. Viola. Robust real-time face detection. ICCV01, 2001.
12. D. W. Jacobs R. Basri. Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern analysis and Machine Intelligence*, 25(2):218–233, 2003.
13. J. Sherrah S. Gong. Fusion of 2d face alignment and 3d head pose estimation for robust and real-time performance. Recognition,analysis and tracking of faces and gestures in real-time systems 1999, 1999.
14. M. La Cascia S. Sclaroff. Fast reliable head tracking under varying illumination. CVPR99, 1999.
15. L. Wiskott. *Labeled Graphs and Dynamic Link Matching for Face Recognition and Scene Analysis*. Verlag Harri Deutsch, 1 edition, 1996.
16. C. Kambhamettu Y. Zhang. Robust 3d head tracking under partial occlusion. Automated Face and Gesture Recognition, 2000.