Robustifying Eye Center Localization by Head Pose Cues

Roberto Valenti[*]
rvalenti@science.uva.nl

Zeynep Yucel
zeynep@ee.bilkent.edu.tr

Theo Gevers
gevers@science.uva.nl

Intelligent Systems Lab Amsterdam
University of Amsterdam
The Netherlands

Abstract

Head pose and eye location estimation are two closely related issues which refer to similar application areas. In recent years, these problems have been studied individually in numerous works in the literature. Previous research shows that cylindrical head models and isophote based schemes provide satisfactory precision in head pose and eye location estimation, respectively. However, the eye locator is not adequate to accurately locate eye in the presence of extreme head poses. Therefore, head pose cues may be suited to enhance the accuracy of eye localization in the presence of severe head poses.

In this paper, a hybrid scheme is proposed in which the transformation matrix obtained from the head pose is used to normalize the eye regions and, in turn the transformation matrix generated by the found eye location is used to correct the pose estimation procedure. The scheme is designed to (1) enhance the accuracy of eye location estimations in low resolution videos, (2) to extend the operating range of the eye locator and (3) to improve the accuracy and re-initialization capabilities of the pose tracker.

From the experimental results it can be derived that the proposed unified scheme improves the accuracy of eye estimations by 16% to 23%. Further, it considerably extends its operating range by more than $15°$, by overcoming the problems introduced by extreme head poses. Finally, the accuracy of the head pose tracker is improved by 12% to 24%.

1. Motivation and Related Work

Due to the swift advances in digital technology, eye location estimation has gained considerable importance in the recent years. Numerous works in the literature studied the development of systems which can estimate eye location for various scenarios. Our aim is to design a method that is capable of doing accurate eye center location and tracking in low resolution videos. In order to be robust, the method should be able to cope with difficult conditions introduced by extreme head poses. There are several methods proposed in the literature for eye center location but their common problem is the use of intrusive and expensive sensors [3] and the sensitivity to head pose variations. While commercially available eye trackers require the user to be either equipped with a head mounted device, or to use a high resolution camera combined with a chinrest to limit the allowed head movement, the methods using image processing techniques are considered to be less invasive and so more desirable in a large range of applications. Furthermore, daylight applications are precluded due to the common use of active infrared (IR) illumination used to obtain accurate eye location through corneal reflection. Non infrared appearance based eye locators [9, 22, 15, 23] can successfully locate eye regions, yet have difficulties in dealing with non frontal-face conditions.

Very promising to our goals is the method proposed in [23]. This method uses isophote (i.e., curves connecting points of equal intensity) properties to infer the center of (semi)circular patterns which represent the eyes. This is based on the observation that the eyes are characterized by radially symmetric brightness patterns. The authors propose a novel center voting mechanism which is used to increase and weight important votes to reinforce the center estimates. The method yields low computational cost allowing real-time processing and it is robust against linear illumination changes and to moderate changes in head pose. However, the accuracy of the eye center location drops significantly in the presence of large head poses. This is due to the fact that the eye structure analyzed is not anymore symmetric and thus the algorithm delivers increasingly poor performance. This observation proves that it is desirable to be able to correct the distortion given by the pose so that the eye structure under analysis keeps the symmetry properties. Assuming that there is a way to compensate for the head pose so that we obtain a normalized image patch on which the eye center

[*] Supported by the MIAUCE European Project (IST-5-0033715-STP).

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

locator is deployed, the results will be improved considerably.

In order to obtain the normalized image patches invariant to changes in head pose, a robust and accurate head pose estimation algorithm should be employed. Throughout the years, different methods for head pose estimation have been developed. The 3D model based approaches achieve robust performance and can deal with large rotations. However, most of the good results are gained in restricted domains, e.g. some systems only work when there is stereo-data available [17, 20], when there is no (self-) occlusion, or when the head is rotating not more than a certain degree [7]. Systems that solve most of these problems, usually do not work in real-time due to the complex face models they use [26]. However, if the face model complexity is reduced to simpler ellipsoidal or cylindrical shape, this creates a prospect for a real-time system. The cylindrical head model (CHM) approach has been used by a number of authors [4, 7, 25]. Among them, the implementation of Xiao *et al.* [25] works remarkably well. This cylindrical approach is capable of tracking the head in situations where the head is rotating far away and as such is a good solution to our problem.

In the literature several approaches have been reported for estimating the head pose, eye location and gaze. The authors of [16, 19] consider a tracking scenario equipped with stereo cameras and employ 2D feature tracking and 3D model fitting. The work proposed by Ji *et al.* [14] describe a real-time eye, gaze and head pose tracker for monitoring driver vigilance. The authors use IR illumination to detect the pupils and derive the head pose by building a feature space from them. Although their compound tracking property promote them against separate methods, the practical limitations and the need for improved accuracy make them less practical in comparison to monocular low resolution implementations.

Very relevant to our work is the approach proposed in [21]. The authors combine a cylindrical head model with an Active Appearance Model (AAM) approach to overcome the sensitivity to large pose variations, initial pose parameters, and problems of re-initialization. Similar to [21], we would like to make use of the competent attributes of the cylindrical head model together with the eye locator proposed in [23] in order to broaden the capabilities of both systems and to improve the precision of each individual component.

To this end, in this paper we propose a unified frame work for head pose and eye location estimation. The head tracker is initialized using the location and orientation of the eyes while the latter are obtained by pose-normalized eye patches obtained from the head tracker. A feedback mechanism is employed in the evaluation of the tracking quality. When the two modules do not yield concurring results, both are adjusted to get in line with each other. In this manner, significant improvement is obtained in the precision of both tracking schemes.

The contributions of this paper are the following:

- Rather than just a sequential combination, we propose a unified framework which provides a deep integration of the CHM pose tracker and the isophote based eye location estimation methods.
- The accuracy of the eye location estimation is improved considerably for the known operating range of the eye locator.
- The working range of the eye locator is extended significantly. The shortcomings of the reported eye locators due to extreme head poses are compensated using the feedback from the head tracker.
- With the help of the obtained eye location, the head tracker provides better pose accuracy and does not lose track of the head in ill cases. Automatic re-initialization of the head tracker with the eye location information eases the recovery of the correct pose.

2. Synergetic Eye Location and CHM Tracking

As mentioned in the previous section, the CHM pose tracker and the isophote based eye location estimation methods have advantages over the other reported methods. However, taken separately, they cannot work efficiently under certain circumstances. In [23] the eye region is assumed to be semi-frontal, hence the eye locator can use curved isophotes to detect circular patterns. The authors claim that the system is robust to slight changes in head pose, but cannot cope with extreme head poses (*e.g.* side faces). On the other hand the CHM pose tracker might erroneously converge to local minima and, after that, might not able to recover the correct track. By integrating the eye locator with the cylindrical head model we aim to obviate these drawbacks.

Instead of a sequential integration of the two systems, in the next section, a deeper integration is proposed by comparing the transformation matrices suggested independently by both systems, as illustrated in Algorithm 1. In this way, the eye locations are detected given the pose, and the pose is adjusted given the eye locations.

2.1. Initialization

The initial parameters of the CHM and its initial transformation matrix are computed as follows: The first time a frontal face is detected, its size is used to initialize the cylinder parameters and the pose $\mathbf{p} = [\omega_x, \omega_y, \omega_z, t_x, t_y, t_z]$ according to anthropometric values [11, 10], where ω_x, ω_y, and ω_z are the rotation parameters and t_x, t_y, t_z are the translation parameters. The eye locations are detected in the face region and are used to give a better estimate of the

Algorithm 1 Pseudo-code of estimating eye locations by head pose

```
Initialize parameters
- Detect face and initialize cylinder parameters
- Get reference eye regions, $R_r$ and $R_l$.
- Use distance between the eyes to get the depth element, $t_z$.
- Initialize pose $\mathbf{p}$ using eye locations

Iterate through all the frames
for $t = 0$ to last frame number do
    - Assume intensity is constant between consecutive frames, $I_{t+1} = I_t$.
    - Compute the gradient $\bigtriangledown I_{t+1}$ and the corresponding Gaussian pyramid for the current frame
    - Initialize pose to the previous pose $p_{t+1} = p_t$
    For all levels of Gaussian Pyramid
    for $l = 0$ to 2 do
        - Calculate motion between two frames $\mathbf{m} = \mathbf{p_{t+1}} * \mathbf{p_t}^{-1}$
        - Load Gaussian pyramid image $I(l)$
        - Initialize $\Delta\vec{p} = [0, 0, 0, 0, 0, 0]$
        while maximum iterations not reached or $\Delta\vec{p} <$ threshold do
            - Transform pixels $p$ of $I(l)$ to $p'$ with transformation matrix $\mathbf{M}$ and parameters $\mathbf{p}$ to compute $I_t(\mathbf{p})$
            - Update and scale face region boundaries $(\vec{u}, \vec{v})$
            - Do ray tracing to calculate $t_z$ for each $p \in (\vec{u}, \vec{v})$
            - Apply perspective projection, $p_x = \vec{u}_n * t_z$, $p_y = \vec{v}_n * t_z$
            - Use inverse motion $\mathbf{m}'$ to get from $p$ to $p'$
            - With back-projection calculate pixels $(u', v')$
            - Compute $I_t$ with $I_{t+1}(\mathbf{m}) - I_t(\mathbf{m}')$.
            - Compute $\bigtriangledown I_{t+1}(\mathbf{m})\frac{\partial \mathbf{T}}{\partial \mathbf{p}}$ where $T$ summarizes the projection model.
            - Compute Hessian matrix in
            - Compute $\sum w \left[ \ \bigtriangledown I_{t+1}\frac{\partial T}{\partial p} \ \right]^T \sum \left[ \ I_t - I_{t+1} \ \right]$
            - Compute $\Delta\vec{p}$ using
            - Update the pose and motion:
            - $\mathbf{p_{t+1}} = \Delta\vec{p} \circ \mathbf{p_{t+1}}$
            - $\mathbf{m} = \Delta\vec{p} \circ \mathbf{m}$
        end while
        - Update transformation matrix $\mathbf{M} = \Delta\vec{p} \circ \mathbf{M}$
        - Transform reference eye regions $R_r$ and $R_l$ using $\mathbf{M}$
        - Remap eye regions to pose normalized view
        - Compute displacements vectors $D$ on pose normalized eye regions accordingly to [23], using,
          $D(x,y) = -\frac{\{\frac{\delta I}{\delta x},\frac{\delta I}{\delta y}\}(\frac{\delta I}{\delta x}^2+\frac{\delta I}{\delta y}^2)}{\frac{\delta I}{\delta y}^2\frac{\delta^2 I}{\delta x^2}-2\frac{\delta I}{\delta x}\frac{\delta^2 I}{\delta x\delta y}\frac{\delta I}{\delta y}+\frac{\delta I}{\delta x}^2\frac{\delta^2 I}{\delta y^2}}.$
        - Vote for centers weighted by $\sqrt{\frac{\delta^2 I}{\delta x^2}^2 + 2\frac{\delta^2 I}{\delta x\delta y}^2 + \frac{\delta^2 I}{\delta y^2}^2}$.
        - Select isocenter closer to the center of eye region as eye estimate
        - Remap eye estimate to cylinder coordinates
        - Create pose vector from eye location and compare it to head tracker's
        if distance between pose vector $>$ threshold then
            average pose vectors and create the new $\mathbf{M}$
        end if
    end for
end for
```

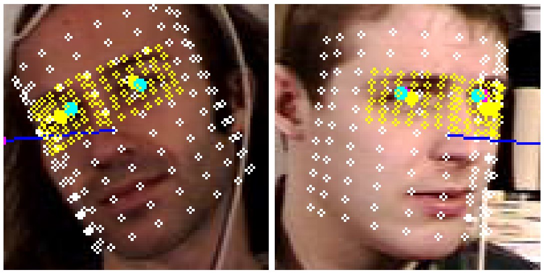

Figure 1. Examples of eye regions sampled by pose (yellow dot meshes)

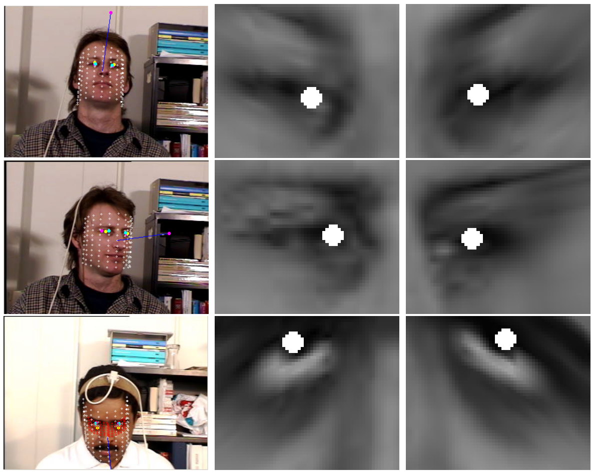

Figure 2. An example of extreme head poses and the respective pose-normalized eye locations. The results of the eye locator in the pose normalized eye region is represented by a white dot.

t_x and t_y. The depth, t_z, is adjusted by using the distance between the detected eyes, d. Finally, since the detected face is assumed to be frontal, the initial pitch (ω_x) and yaw (ω_y) angles are assumed to be null, while the roll angle (ω_z) is initialized by the relative position of the eyes.

After the cylinder is initialized in the 3D space, the 2D eye locations detected in the first frame are used as reference points (*e.g*. the yellow "+" markers in Figure 3). These reference points are projected onto the cylindrical head model, so that the depth values of the eye locations are known. The reference eye points are then used to estimate the successive eye locations and are in turn updated by using the average of the found eye locations.

2.2. Eye Location by Pose Cues

Around each reference point projected onto the 3D model, an area is sampled and transformed by using the transformation matrix obtained by the head pose tracker (Figure 1). The pixels under these sampled points are then remapped into a normalized canonical view (Figure 2). Note from Figure 2 that extreme head poses are also successfully corrected, although some perspective projection

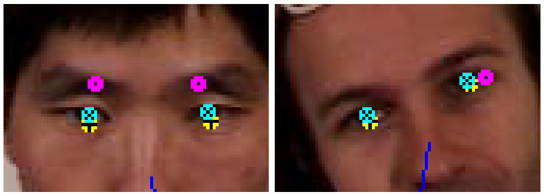

Figure 3. A mistake of the standard eye locator (.), corrected by the pose cues (x) according to the reference points (+)

errors are retained. A modified version of the eye locator proposed in [23] is then applied to these pose normalized eye regions: Instead of using the meanshift algorithm [8] to estimate the area with the maximum density of votes as proposed by [23], the highest peak in the centermap which is closer to the center of the eye region (therefore closer to the reference eye location obtained by pose cues), is selected as estimated eye center (the white dots in Figure 2 and the ”x” markers in Figure 3) . In this way, as long as the CHM tracker is correctly estimating the head pose, the localized eyes can be considered to be optimal. Figure 3 shows two examples in which the default eye locator would fail (”.” marker) but the pose normalized eye estimation would be correct (”x” marker).

2.3. Pose Estimation by Eye Location Cues

Since there is no certainty about the quality of the pose obtained by the head tracker, the found pose-normalized eye location can be used as a tool for quality control. Given that the 3D position of the eyes is known, it is possible to calculate its pose vector and compare it with the one obtained by the head tracker. When the distance between the two pose vectors is larger than a certain threshold, the vectors are averaged and the transformation matrix of the tracker is recomputed. In this way the head model is adjusted to a location that should ease the correct convergence and therefore recover the correct track. As an additional quality control, the standard eye locator in [23] is constantly used to verify that the eye location found by pose cues is consistent with the one obtained without pose cues. Therefore, as in [18], when trustable evidence (*e.g.* the eye location in a frontal face) is collected and found to be in contrast with the tracking procedure, the latter is adjusted to reflect this evidence.

In this manner, the eye locations are used to both initialize the cylinder pose and update it in case it becomes unstable, while the pose normalized eye locations are used to constantly validate the tracking process. Therefore, the CHM tracker and the eye locator interact and adjust their own estimations by using each other’s information. This synergy between the two systems allows for an initialization-free and self-adjusting system. A schematic overview of the full system is shown in Figure 4.

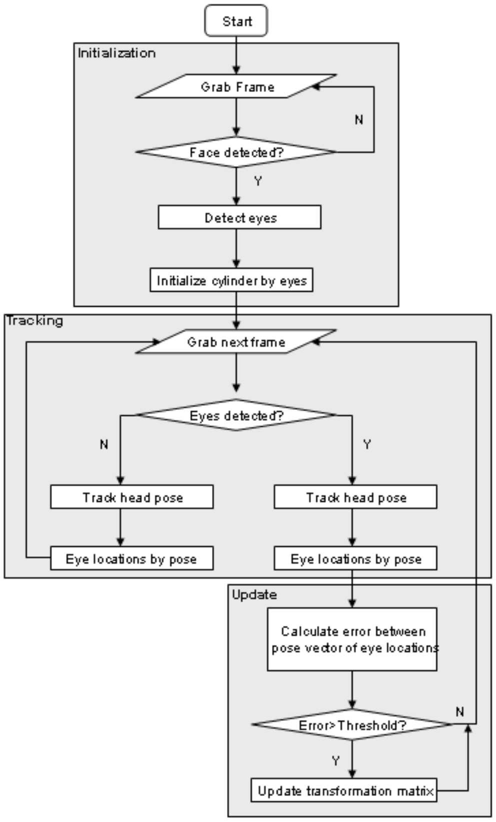

Figure 4. A schematic diagram of the components of the system

3. Experiments

The aim of our experiments is to analyze the improvement obtained on the eye locator given the pose, and on the pose given the eye locations. Unfortunately we are not aware of any combined head pose and eye location video dataset available in the literature. Therefore we selected a freely available database with ground truth of the head pose and have manually annotated the eyes of the subjects on 9000 frames.

3.1. Dataset

The Boston University head pose database [6] is employed in the experiments. The database consists of 45 video sequences, where 5 subjects were asked to perform 9 different head motions under uniform illumination in a standard office setting. The head is always visible and there is no occlusion except for some minor self-occlusions. Note that the videos are in low resolution (320 × 240 pixels) and the iris diameter roughly corresponds to 4 pixels.

A Flock of Birds tracker records the pose information coming from the magnetic sensor on the person’s head. This system claims a nominal accuracy of 1.8 mm in translation and 0.5 degrees in rotation. However, Cascia *et al.* [6] have

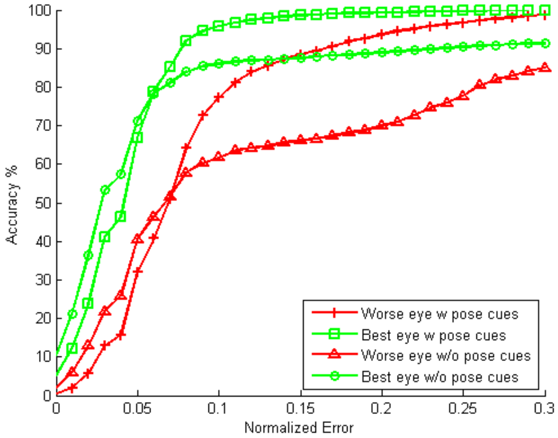

Figure 5. A comparison between the eye detection results with and without pose cues

experienced a lower accuracy due to the interfering electromagnetic noise in the operating environment. Nonetheless, the stored measurements are still reliable enough to be used as ground truth.

3.2. Error Measures

In quantifying the eye location error, we used the 2D *normalized error*. This measure was introduced by Jesorsky *et al.* [13] and is widely used in eye location literature [2, 5, 12, 22, 27]. The normalized error represents the error obtained by the worse eye estimation and is defined as:

$$e = \frac{\max(d_{left}, d_{right})}{d}, \tag{1}$$

where d_{left} and d_{right} are the Euclidean distance between the located eyes and the ones in the ground truth, and d is the Euclidean distance between the eyes in the ground truth. In this measure, $e \leq 0.25$ (a quarter of the interocular distance) corresponds roughly to the distance between the eye center and the eye corners, $e \leq 0.1$ corresponds to the range of the iris, and $e \leq 0.05$ corresponds the range of the cornea. In order to give upper and lower bounds to the accuracy, in Figure 5 we also show the *minimum normalized error*, obtained by considering the best eye estimation only.

For the pose estimation error, the root mean square error (RMSE) and standard deviation (STD) values are used for the three planar rotations: ω_x, ω_y and ω_z.

3.3. Results

Here we elaborate on the performance of the eye localization and head tracking accuracy. In Section 3.3.1 the accuracy for certain normalized error values are presented, where Section 3.3.2 gives RMSE and STD for head tracking error and provides a comparison to two other methods, which employ the same dataset.

Table 1. Effect of pose cues in eye localization

	Worse eye		Best eye	
	Without pose	With pose	Without pose	With pose
$e \leq 0.05$	40.6	31.93	66.78	71.27
$e \leq 0.1$	61.73	77.27	86.03	95.81
$e \leq 0.15$	66.14	88.46	87.87	98.6
$e \leq 0.2$	70	93.67	93.67	99.29
$e \leq 0.25$	77.72	96.74	96.74	99.73

3.3.1 Eye Location Estimation

In order to illustrate the effect of pose cues in eye location estimation, the accuracy of our unified approach is presented in Figure 5 together with the accuracy of the standard eye locator [23]. In the latter, the approximate face position is estimated using the boosted cascade face detector proposed by Viola and Jones [24], where the rough positions of the left and right eye regions are estimated by anthropometric relations [11]. For the cases in which the face cannot be detected, the maximum possible localization error is assigned considering the limits of the detected face and anthropometric measures in the following manner. The maximum achievable error is assumed to be half of the interocular distance, which corresponds to 0.5. Therefore a default error value of 0.5 is assigned to both eyes for the frames in which a face is not detected. In our experiments a face was not detected in 641 frames, which corresponds to 7.12% of the full dataset. The working range of the face detector is around $30°$ around each axis, while certain head poses in the dataset are larger than $45°$.

The accuracy is represented in percentages for a normalized error of range $[0, 0.3]$. A performance comparison is provided for the best and worse eye location estimations, where certain precise values are also given in Table 1 for several normalized error values.

From Figure 5 it is clear that the pose cues improve the overall accuracy of the eye detector. In fact, for an allowed error bigger than 0.1, the unified scheme provides an improvement in accuracy from 16% to 23%. For smaller error values, the system performs slightly worse than the standard eye locator. The reasons behind this decrease in accuracy will be discussed in Section 3.4.

3.3.2 Head Pose Estimation

In the determination of pose accuracy, two scenarios are considered: in the first scenario the template is created from the first frame of the video sequence and is kept constant for the rest of the video; in the second scenario the template is updated at each frame, so that the tracking is always performed between two successive frames. Table 2 shows the improvement in RMSE and STD given by using eye location cues in both scenarios. Note that, without using the eye cues, the updated template gives the best results. On

Table 2. Comparison of RMSE and STD

	Fixed template				Updated template				Sung	An
	With eye cues		Without eye cues		With eye cues		Without eye cues		*et al.* [21]	*et al.* [1]
	RMSE	STD	RMSE	STD	RMSE	STD	RMSE	STD	RMSE	RMSE
Pitch (ω_x)	5.26	4.67	6.00	5.21	5.57	4.56	5.97	4.87	5.6	7.22
Yaw (ω_y)	6.10	5.79	8.07	7.37	6.45	5.72	6.40	5.49	5.4	5.33
Roll (ω_z)	3.00	2.82	3.85	3.43	3.93	3.57	4.15	3.72	3.1	3.22

the other hand, if the eye cues are considered, the accuracy of the fixed template becomes better than the updated one. This due to the fact that by using the eye cues while updating the template might introduce some errors at each update, which cannot be recovered at later stages. However, for both scenarios, the use of eye cues presents an improvement in estimation of the pose angles.

In the last two columns of Table 2, we compare our results with two other works in the literature, which use the same database. Similar to our method, Sung *et al.* [21] propose a hybrid approach combining active appearance models and cylinder head models to extend the operating range of AAM. An *et al.* [1] propose to replace the traditional CHM with a simple 3D ellipsoidal model. They provide comparison of accuracy with planar and cylindrical models. Here we consider the accuracy reported by Sung *et al.* and from An *et al.* on the cylindrical head model [1]. From Table 2, it is obvious that our method provides comparable or better results with respect to the compared methods.

3.4. Discussion

The eye detection results obtained by using pose cues depict a significant overall improvement over the baseline results. However, as shown in Section 3.3, we note a small drop in accuracy for precise eye location ($e \leq 0.05$). We believe this is due to interpolation errors occurring while sampling and remapping the image pixels to pose-normalized eye regions. In fact, as seen in Figure 2, in specific extreme head poses the sampled eye might not result in a completely circular shape due to perspective projections, therefore the detection might be shifted by one or two pixels. Given the low resolution of the videos, this shift can easily bring the detection accuracy over the $e \leq 0.05$ range. However, an error at the given low resolution is barely noticeable.

Regarding the improvement obtained on the pose estimation, our experiments also show that using eye cues has an overall positive effect on the average RMSE. However, it is important to note that by enhancing the head tracker using the eye cues to fix the transformation matrix does not have a direct effect on the accuracy. The main effect is obtained by the re-initialization of the cylinder in a position which allows for a correct convergence once the pose tracker converges to a local minimum. In fact, by closely analyzing our results we notice that by using the eye cues the accuracy of the pose was decreased for particular subjects showing extreme head poses.

This issue is related to the approach used to fix the transformation matrix. In our approach we assume that the eye located given the correct pose are the correct ones, but this will not be true in the presence of highlights, closed eye or very extreme head poses (*e.g.* when the head is turned by 90° and only one eye is visible). In these specific cases, averaging by the transformation matrix suggested by the eye location might negatively affect an otherwise correct transformation matrix given by the head tracker. Fortunately the eye locator can be considered quite accurate and therefore these cases do not occur very often, and the track is immediately recovered as soon as the difficult condition is resolved or a semi-frontal face is detected again.

4. Conclusions

In this paper we propose a deep integration of a CHM based head tracker and isophote based eye locator in a complementary manner, so that both system can benefit from each other's evidence. Experimental results prove that the accuracy of both independent systems is significantly improved by their blending. Numerically, the eye location estimation of the unified scheme achieves an improvement in accuracy from 16% to 23%, while the pose error is improved from 12% to 24%. Besides the improvements in accuracy, the operating range of the eye locator is significantly extended (more than 15°) by the head tracker and the ineffectiveness of the previously reported eye location methods against extreme head poses is compensated. Furthermore, automatic quality control and re-initialization of the head tracker is provided by the integration of the eye locator, which helps the system in recovering to the correct pose in ill cases. Consequently, the proposed unified approach allows for an autonomous and self-correcting system for head pose estimation and eye localization.

References

[1] K. H. An and M. Chung. 3d head tracking and pose-robust 2d texture map-based face recognition using a simple ellipsoid model. In *Intelligent robots and Systems*, 2008.

[2] L. Bai, L. Shen, and Y. Wang. A novel eye location algorithm based on radial symmetry transform. In *ICPR*, pages 511–514, 2006.

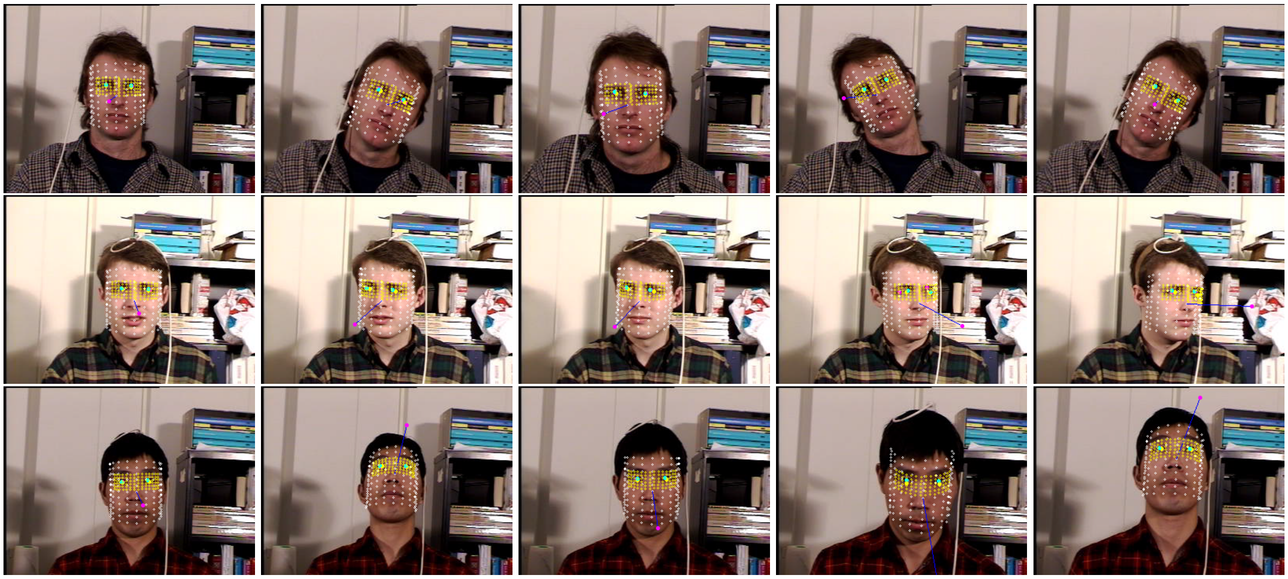

Figure 6. Qualitative examples of result on roll, yaw and pitch angles on videos showing extreme head poses.

[3] R. Bates, H. Istance, L. Oosthuizen, and P. Majaranta. Survey of de-facto standards in eye tracking. In *COGAIN Conf. on Comm. by Gaze Inter.*, 2005.

[4] L. Brown. 3D head tracking using motion adaptive texture-mapping. In *CVPR*, 2001.

[5] P. Campadelli, R. Lanzarotti, and G. Lipori. Precise eye localization through a general-to-specific model definition. In *BMVC*, 2006.

[6] E. L. Cascia, S. Sclaroff, and V. Athitsos. Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3d models. *Pattern Analysis and Machine Intelligence*, 22(4), 2000.

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

[8] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *PAMI*, 25(5):564–577, 2003.

[9] D. Cristinacce, T. Cootes, and I. Scott. A multi-stage approach to facial feature detection. In *BMVC*, pages 277–286, 2004.

[10] N. A. Dogson. Variation and extrema of human interpupillary distance. In *SPIE*, 2004.

[11] C. C. Gordon, B. Bradtmiller, T. Churchill, C. E. Clauser, J. T. McConville, I. O. Tebbetts, and R. A. Walker. Anthropometric survey of us army personnel: Methods and summary statistics. Technical report, United States Army Natick Research, 1988.

[12] M. Hamouz, J. Kittlerand, J. K. Kamarainen, P. Paalanen, H. Kalviainen, and J. Matas. Feature-based affine-invariant localization of faces. *PAMI*, 27(9):1490–1495, 2005.

[13] O. Jesorsky, K. J. Kirchbergand, and R. Frischholz. Robust face detection using the Hausdorff distance. In *Audio and Video Biom. Pers. Auth.*, pages 90–95, 1992.

[14] Q. Ji and X. Yang. Real-time eye, gaze and face pose tracng for monitoring drivr vigilance. In *Real Time Imaging*, 2002.

[15] B. Kroon, S. Boughorbel, and A. Hanjalic. Accurate eye localization in webcam content. In *FG*, 2008.

[16] Y. Matsumoto and A. Zelinsky. An algorithm for real-time stereo vision implementation of headpose and gaze direction masurement. In *FG*, 2000.

[17] L. Morency and T. Darrell. Head gesture recognition in intelligent interfaces: The role of context in improving recognition. In *IUI*, 2006.

[18] L. Morency, A. Rahimi, and T. Darrell. Adaptive view based apperance models. In *CVPR*, 2003.

[19] R. Newman, Y. Matsumoto, S. Rougeaux, and A. Zelinsky. Real-time stereo tracking for head pose and gaze estimation. In *Automatic Face and Gesture Recognition*, 2000.

[20] D. Russakoff and M. Herman. Head tracking using stereo. *Machine Vision and Applications*, 13:164–173, 2002.

[21] J. Sung, T. Kanade, and D. Kim. Pose robust face tracking by combining active apperance models and cylinder head models. *International Journal of Computer Vision*, 80(2), 2008.

[22] M. Türkan, M. Pardás, and A. Çetin. Human eye localization using edge projection. In *Comp. Vis. Theory and App.*, 2007.

[23] R. Valenti and T. Gevers. Accurate eye center location and tracking using isophote curvature. In *CVPR*, 2008.

[24] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.

[25] J. Xiao, T. Kanade, and J. Cohn. Robust full motion recovery of head by dynamic templates and re-registration techniques. In *FG*, 2002.

[26] Y. Zhang and C. Kambhamettu. 3D head tracking under partial occlusion. *Pattern Recogn.*, 35:1545–1557, 2002.

[27] Z. H. Zhou and X. Geng. Projection functions for eye detection. In *Pattern Recognition*, pages 1049–1056, 2004.