This paper appeared in the Gesture Recognition, and Kinect competition Workshop, CVPR 2012

Gaze Estimation from Multimodal Kinect Data

Kenneth Alberto Funes Mora and Jean-Marc Odobez
Idiap Research Institute, CH-1920, Martigny, Switzerland
École Polytechnique Fédéral de Lausanne, CH-1015, Lausanne, Switzerland
{kfunes,odobez}@idiap.ch

Abstract

This paper addresses the problem of free gaze estimation under unrestricted head motion. More precisely, unlike previous approaches that mainly focus on estimating gaze towards a small planar screen, we propose a method to estimate the gaze direction in the 3D space. In this context the paper makes the following contributions: (i) leveraging on Kinect device, we propose a multimodal method that rely on depth sensing to obtain robust and accurate head pose tracking even under large head pose, and on the visual data to obtain the remaining eye-in-head gaze directional information from the eye image; (ii) a rectification scheme of the image that exploits the 3D mesh tracking, allowing to conduct a head pose free eye-in-head gaze directional estimation; (iii) a simple way of collecting ground truth data thanks to the Kinect device. Results on three users demonstrate the great potential of our approach.

1. Introduction

Understanding human behaviour or intentions is a central issue in numerous applications. At the heart of this issue lies, amongst others, the difficulty of sensing human behaviours in an accurate way, i.e. the challenge of developing algorithms that can reliably extract subtle human characteristics -e.g. body gestures, facial expressions, emotion- that allow a fine analysis of behaviour.

One such characteristic of interest is the gaze. It indicates where and what a person is looking at, and conveys a wealth of information about that person: what is he interested in, what is he doing, how does he explore a new environment or react to different visual stimuli. Gaze plays also an important role in face-to-face conversations and more generally group interaction, as shown in a large body of social psychology studies [8], with functions such as establishing relationships, regulate the course of interaction, expressing intimacy, or exercising social control [7]. Thus, automatic gaze tracking tools are or would be useful in numerous applications: facilitating the annotation work of psychologists, communication experts and sociologists allowing them to study larger corpus; marketing analysis through the retrieval of user's reaction to a product; assisting tools for disabled people lacking certain ways of communication; focused object dependent video coding [10]; and finally, intuitive interfaces for Human-Computer Interaction (HCI) and Human Robotic Interactions (HRI).

Over the last 30 years, researchers have proposed gaze estimation strategies for diverse setups [5], from which commercial systems have emerged. However most of these methods involve specialized hardware whose high cost reduces the availability to the general public, or involve controlled scenarios, like fixed head pose, restricted head motion, or looking at a planar screen. Therefore there is still the need for a gaze estimation system with minimal intrusion, hardware, user cooperation, calibration and cost.

This paper addresses some of these issues by leveraging on the multimodal Microsoft Kinect sensor. More precisely, we exploit the depth sensor to perform an accurate tracking of a 3D mesh model and robustly estimate a person head pose. In a second step, thanks to the use of the image modality, a simple eye region rectification step, and training data easily collected via the Kinect, we compute a person's eye-in-head gaze direction, which can in turn simply be added to the head pose to generate the final gaze estimate.

In this way, despite the current lack of certain processing steps that could certainly improve the results, like better eye stabilization, we obtain a gaze tracking algorithm that can be used with unrestricted head motion and in more open conditions than in previous works. This paves the way to gaze estimation systems working in more open spaces.

In the rest of the paper, Section 2 describes related works, Section 3 introduces our method, Section 4 our strategy to easily obtain training data using the Kinect. Results are given in Section 5 and Section 6 concludes the work.

2. Related work

A plethora of methods have been proposed for gaze estimation [5]. Its analysis has been dominantly investigated for Human-Computer Interfaces (HCI) applications, due to its potential in the development of intuitive interfaces.

Recently there has been an increased interest in *appearance* based methods that learn a direct mapping from the high dimensional eye images to the low dimensional space of gaze coordinates without the need to explicitly extract features (pupil, eye corners) which are difficult to obtain from low-resolution images. Such mapping can be modeled using regression support vector machines [11], localized linear regression [3, 4, 13] or gaussian processes [15].

One of their main challenges is the variation of eye appearance due to head-pose. This has been addressed by requesting a fixed head position [3] or using specialized head mounted hardware [11]. Few works address head pose invariance. For the method described in [13] a large quantity of data was collected to have samples with head pose-gaze coordinates variability, but this implies a time consuming training step. Learning a gaze model for a fixed given head pose and then correcting the infered gaze vector for the head-pose estimated online was proposed in [4]. However, the method is constrained to estimate gaze in a 2D planar surface and the addressed range of head poses is still low.

Due to the dependency of the eye image appearance to head-pose, estimating this pose becomes critical for gaze tracking. To this end, many methods have been proposed [9], some of which benefit from multimodal data, and in particular depth imaging combined with standard video. Regression methods have been proposed to infer head pose parameters directly from the depth image [2]. Even though good generalization across individuals has been achieved, semantic information, such as the location of the eyes, is lost.

If both modalities are calibrated, 3D representations of the scene can be obtained from depth imaging. With this explicit representation, the Iterative Closest Points (ICP)[16] method can be used to infer the pose change between successive frames. Either the previous frame is registered to the current one, or a template mesh is created off-line, and registered to each frame [14]. The advantage of using a template mesh is its predefined semantic interpretation. This characteristic is exploited by our method which is now explained in more detail.

3. Proposed method

The different steps of our method are shown in Fig. 1. Given a learned person-specific 3D mesh model (Section 3.1), the method works by first estimating the head pose from the depth data, as explained in Section 3.2. Then, using the estimated head pose and the 3D mesh, we map back the head image into a frontal pose and crop the resulting image around each eye. In this way we are able to estimate the gaze vectors from the eye-in-head images (Section 3.4), that is, estimate the eye gaze in the reference system of the head. Finally we transform back the gaze direction to the world coordinate system. The following sections describe in detail each one of the steps.

3.1. Face model learning

We decided to rely on a 3D Morphable Model (3DMM) to generate person specific 3D face templates. These models (3DMM) span a large set of facial shapes using a small set of coefficients. For our task we used the Basel Face Model (BFM) [12], which is a rich 3DMM for human faces, built from a large group of individuals. It has a high mesh density (53,490 vertices) and includes the face, frontal neck and ears. The 3DMM shape can deform according to Eq. 1, where $\mathbf{x}$ (the model instance) denotes the set of 3D vertices coordinates $\{v_i\}$ stacked as a large column vector, μ is the mean shape and $\mathbf{M}$ is the shape basis. The model parameter is the vector α.

$$\mathbf{x}(\alpha) = \mu + \mathbf{M}\alpha \tag{1}$$

The topology is kept fixed for all the head instances. This is a main advantage of using 3DMMs to generate face models, as semantic information is predefined in the topology, and is kept across individuals.

To generate person-specific 3D meshes, we fit the 3DMM to depth data. This is done by registering the 3DMM, whose shape deformations are constrained by α, to the reconstructed mesh from the Kinect. To solve this problem we used the method from [1], and minimize the following cost function:

$$E(\mathbf{X}) = E_d(\mathbf{X}) + \gamma E_l(\mathbf{X}) + \lambda E_s(\mathbf{X}) \tag{2}$$

where the parameters $\mathbf{X} = \{\mathbf{R}, \mathbf{t}, \alpha\}$ to optimize are those of a rigid transformation $(\mathbf{R}, \mathbf{t})$ and of the 3DMM coefficients α, and the different terms are given as follows:

$$\begin{aligned} E_d(\mathbf{X}) &:= \sum_i w_i \|\mathbf{R}(\mu_i + \mathbf{M}_i\alpha) + \mathbf{t} - u_i\|^2 \\ E_l(\mathbf{X}) &:= \sum_{(i,\mathbf{l})\in L} \|\mathbf{R}(\mu_i + \mathbf{M}_i\alpha) + \mathbf{t} - \mathbf{l}\|^2 \\ E_s(\mathbf{X}) &:= \|\alpha\|^2 \end{aligned} \tag{3}$$

in which μ_i and $\mathbf{M}_i$ represent the rows corresponding to the vertex i in μ and $\mathbf{M}$. The data term E_d represents the distance of each deformed and rigidly transformed vertex, v_i, to its closest correspondent u_i[1] in the target mesh. Each individual term is weighted by a weight w_i deduced from a robust function and which reduces the influence of pairs with a large distance (outliers). Additionally the weights are set to zero for pairs whose corresponding points are in the border of the target mesh or have a large angle between surface normals. The term E_l models the same quantity but for a set of known corresponding landmarks L. Finally, the

[1] The corresponding points are computed at each iteration as the closest point with constraints, see the head tracking section

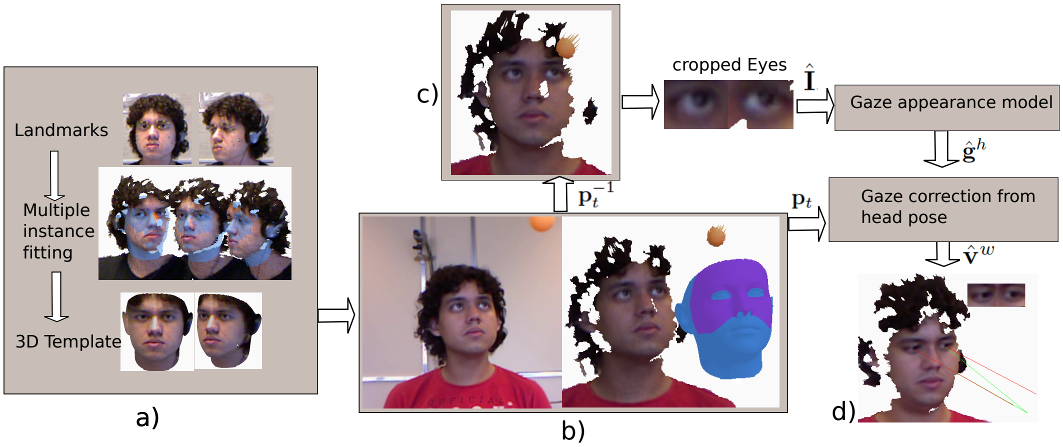


Figure 1: Proposed method pipeline. a) Offline step. From multiple 3D face instances the 3DMM is fit to obtain a person specific 3D model. b)-d) Online steps. b) The person model is registered at each instant to multimodal data to retrieve the head pose. In the figure, the model is rendered with a horizontal spacing for visualization. The region used for tracking is rendered in purple. c) Head stabilization computed from the inverse head pose parameters and 3D mesh, creating a frontal pose face image. Further steps show the gaze estimation in the head coordinate system. The final gaze vector is corrected according to the estimated head pose. d) Obtained gaze vectors (in red our estimation and in green the ground truth).

regularization term E_s foster the estimation of small values for α. This term is weighted by the *stiffness* parameter λ in Eq.2, which controls how much the template mesh can deform. The algorithm works as follows:

- Initialize $\mathbf{X}^0$. Then, for each stiffness value $\lambda^i \in \{\lambda^1, \ldots, \lambda^n\}, \lambda^i > \lambda^{i+1}$, do:
  - Until $\|\mathbf{X}^j - \mathbf{X}^{j-1}\| < \epsilon$:
    - ∗ Find correspondences u_k in the target surface for each point v_k in the template.
    - ∗ Determine $\mathbf{X}^j$ minimizing Eq. 2 using λ^i.

Multi-instance 3DMM fitting. We want to build personalized head templates from Kinect data. However these devices present high levels of noise and missing information due to occlusions and a large field of view. To address this we fitted the 3DMM to a set of instances of the target face, with different head poses. We extended the 3DMM fitting method by having sets of equations for each of the target instances. Each instance had its own rotation and translation parameters but all were sharing the α parameters. The advantage is to obtain a template in a single step.

A current drawback is the need for a manual placement of landmarks, although this can take less than 30s per instance. Our implementation takes ≈2 minutes for the model fitting. This can be much faster as our implementation is not optimized and we use the (unnecessary) full resolution of the BFM. An example is shown in Fig. 1a.

3.2. Head pose tracking

During the online part, the first step of our system is to estimate the head pose. Inspired by the work from [14] we built a 3D face tracker based on video and depth data from the Kinect. The algorithm is based on the Iterative Closest Points (ICP) algorithm using point-to-plane constraints and the personalized template. It re-estimates the pose parameters at a given frame initializing ICP from the inferred values in the previous frame. Tracking is obtained by repeating this process frame by frame. At each time t we obtain the head pose parameters as $\mathbf{p}_t = \{\mathbf{R}_t, \mathbf{t}_t\}$, i.e. the head rotation and translation.

The overall initialization is done with a standard frontal face detector in the visual image. From depth data, the set of 3D points, within the detection bounding box, are used to initialize the translation. The head orientation is assumed to be frontal. A smaller segment of the personalized 3D model is used by ICP to avoid mismatches due to non-rigid deformations as done in [14]. This step is illustrated in Fig. 1b.

3.3. Head stabilization

Given that we represent the Kinect data as a textured 3D mesh, we can render the scene using the inverse rigid transformation of the head pose parameters, i.e. $\mathbf{p}_t^{-1} = \{\mathbf{R}_t^\top, -\mathbf{R}_t^\top \mathbf{t}_t\}$. In this manner we obtain, at each frame, a frontal image of the face as shown in Fig. 1c.

The localization of the eyes is predefined in the mesh topology. We use this information to crop eye-images in the rendered face image. These eye images will be the input to the gaze estimation method.

3.4. Eye-in-Head Gaze estimation

Our main objective is to estimate the gaze direction under free-head motions. However, due to the head stabilization approach, we reduced this problem to a frontal head pose gaze estimation. Thus, our task consist in building an eye appearance model from which we can estimate gaze.

In this section we assume we have a set of pairs of eye images and gaze vectors $\{(\mathbf{I}_i, \mathbf{g}_i^h)\}$ covering the gaze space (in Section 4 we describe a simple method to collect these samples). In the reference system of the head, the gaze is parametrized by the angles $\mathbf{g}_i^h = (\phi_i, \theta_i)$. Where θ is the gaze elevation, and ϕ is the gaze yaw.

The image descriptor $\mathbf{e}_i$ is computed by dividing the eye image in a grid of $r \times c$. At each bin $j = (r, c)$ of the grid, the sum of pixel intensity $S_j$is computed. The descriptor is then the concatenation of all the S_j, and it is normalized such that $\sum_j e_i^j = 1$. Thus, the appearance model consist of the set $\{(\mathbf{e}_i, \mathbf{g}_i^h)\}$. We used $r = 3$ and $c = 5$, following the approach from [3].

Given a test image $\hat{\mathbf{I}}$, with descriptor $\hat{\mathbf{e}}$, we want to infer its gaze direction $\hat{\mathbf{g}}^h$. We followed the method in [3]. The goal is to obtain the weights w_i which reconstruct best the test image from the convex combination of the samples in the appearance model. We then use these weights to combine the gaze parameters as $\hat{\mathbf{g}}^h = \sum_i w_i \mathbf{g}_i^h$. Denoting by E the matrix whose column i correspond to $\mathbf{e}_i$, this problem is formulated in Eq. 4 as

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} ||\mathbf{w}||_1 \quad s.t. \quad ||E\mathbf{w} - \hat{\mathbf{e}}||_2 < \epsilon \tag{4}$$

This method, that will be referred to as adaptive linear regression (ALR), imposes sparsity over the solution of w_i. Its success depends on $\{\mathbf{e}_i\}$ being sparse enough [3].

3.5. 3D gaze estimation

The estimated gaze direction $\hat{\mathbf{g}}^h$ in the pose-corrected frontal head reference can be transformed into a gaze direction $\hat{\mathbf{v}}^w \in \mathbb{R}^3$ in the world coordinate system as follows. Note that the angular gaze parameters $\hat{\mathbf{g}}^h$ can be represented by the unit vector $\hat{\mathbf{v}}^h \in \mathbb{R}^3$ pointing in the direction indicated by (ϕ, θ). Thus, $\hat{\mathbf{v}}^w$ can be computed using the estimated head pose parameters $\mathbf{p}_t = (\mathbf{R}_t, \mathbf{t}_t)$ as $\hat{\mathbf{v}}^w = \mathbf{R}_t \hat{\mathbf{v}}^h$. The predefined center of the eye is transformed by the same $\mathbf{p}_t$.

4. Gaze ground-truth collection

4.1. Proposed setup and methodology

Here we propose a method to collect ground truth data from a Kinect device. Under free-head movements the gaze vector $\mathbf{v}^w$ was defined as the *visual axis*, i.e. the vector which points from the eye fovea to the visual target.

Fig. 2 shows the proposed setup. The system includes a Microsoft Kinect device and a discriminative small object. In our experiments we used a 4cm orange ball. The participant is requested to follow the target with the eyes while the target is moving. Given that the gaze target is discriminative, in both color and depth, we can reliably track its 3D position at each moment.

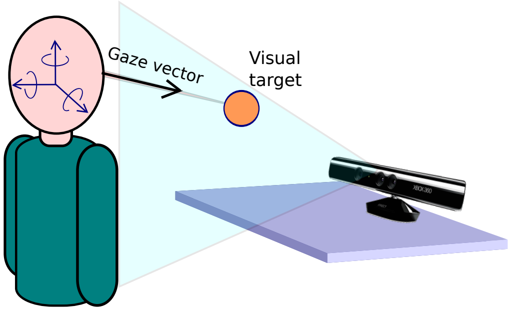


Figure 2: Proposed experimental setup.

Using the head pose tracker, we approximate the visual axis as the vector from the eye-ball center, predefined in the topology, to the estimated location of the visual target. This is done for both eyes. The drawback of this method is that it inherits errors from the head pose tracker, there is uncertainty introduced by the size of the target, and the eye centers is defined approximately. However, it is clearly advantageous as it provides a simple way to collect a large corpus of labeled data from which to train and test our methods.

4.2. Recorded data

Using the proposed setup we collected videos of 3 participants following a target with the eyes. For each of them a personalized head 3D head template was learn using the method described in Section 3.1. In addition, in order to reconstruct the scene as a textured 3D mesh we first calibrated the Kinect sensor using [6].

Each recording was divided in two parts. In the first part the participant was asked to keep a frontal head pose while following the target. It was not necessary for the participant to keep a strict frontal head pose, as we assume the head pose stabilization is accurate for near frontal head poses. The purpose is to gather frontal eye images without occlusions to build appearance based gaze models.

In the second part of the recording the participant was asked to perform free head rotations and translations while keeping track of the visual target with the eyes. The participants went through highly challenging head poses ranging yaw angles of up to ±60° and pitch values up to ±50°.

For the creation of the appearance model, we used the first half of the part where the participant had a frontal head pose. Instead of taking all the samples to build the appearance model, we defined a grid of 42 points in the gaze space by dividing the intervals of $[-40°, 40°]$ for θ and $[-50°, 50°]$ for ϕ into 6 θ values and 7 ϕ values.

We can automatically select samples from the recording with gaze directions close to the grid points. In our experiments, although the fixed head session recordings is of about 1.5 minutes, the appearance model can be created fast as only a few samples are needed. Fig. 3a) shows the computed gaze parameters for a complete recording session (frontal head pose), while Fig. 3b) shows a grid obtained from this recording session.

5. Experiments

We conducted a series of experiments to validate our method. The head pose tracker was validated using the BIWI head pose database [2] and we found deviations of around 1° with respect to their reported ground truth (which is obtained using a similar head pose tracker).

As it was mentioned in Section 4, the gaze appearance model is created from the first half of the session for a frontal head pose. We used the second half as test samples, together with all the samples for the free head pose session.

The error measure is given by the angle κ, which lies between the estimated gaze vector $\hat{\mathbf{v}}$ and the ground truth $\mathbf{v}^g$. The origin point for these vectors is common: the predefined eye center. This measure is independent of the head pose.

Besides the method described in Section 3.4 (ALR) to estimate gaze, we also compared to a nearest neighbors (NN) and k-Nearest Neighbors (kNN) approach. The search for NN and kNN is done within the appearance model.

Result discussion. Table 1 shows the obtained results. For the frontal session it is clear that kNN and ALR are more accurate than a simple NN approach. The results given by kNN and ALR are similar. The obtained results are satisfactory for the frontal case.

We argue that the main sources of these errors are varied: low-resolution eye images, the uncertainty introduced by the target object, the current eye image representation (cf. Section 3.4) and some jitter introduced by our head tracker. Remember that the only rectification step that is applied comes from the head shape model estimated globally. We expect a fine local stabilization of the eye rectified image as done in other works [3] to greatly improve accuracy.

Interestingly, for the free head movements session, although the participants go through large head poses, the errors are only slightly above those of the frontal head pose,

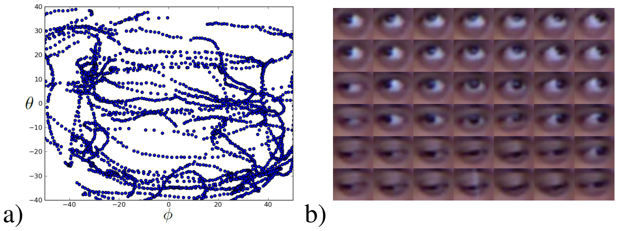
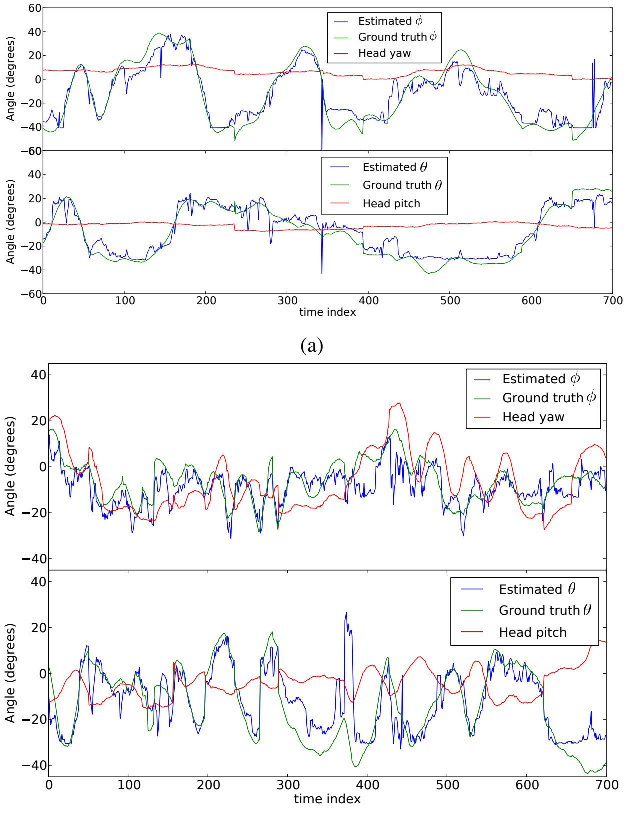


Figure 3: a) Distribution of the gaze points obtained for a recording session (frontal head pose) b) The appearance model constructed from it.

Table 1: Gaze estimation error κ for both test sessions. Reporting mean/median error in degrees.

		Method		
Participant	eye	NN	kNN (k=5)	ALR
1 (frontal)	left right	8.0/7.1 8.0/7.0	7.6/6.0 6.9/5.8	7.6/6.9 7.6/6.3
2 (frontal)	left right	14.4/9.9 12.6/9.2	12.3/9.1 10.8/7.5	11.1/7.8 12.6/7.9
3 (frontal)	left right	12.7/11.0 12.3/10.0	10.8/9.5 11.0/9.3	10.5/9.0 11.3/9.6
1 (free)	left right	- -	12.8/10.9 11.0/8.8	10.8/8.5 11.2/8.0
2 (free)	left right	- -	24.8/19.4 22.0/16.6	25.1/18.0 22.8/16.3
3 (free)	left right	- -	16.3/14.1 10.9/9.0	10.8/8.7 8.6/6.8



(a)

(b)

Figure 4: Estimated gaze vs. time for the right eye of participant number 3. Showing both gaze yaw and elevation together with the head yaw and pitch values. a) Frontal session interval; b) Free head pose session interval.

except for participant 2. In this case, however, the person goes through extreme head poses ($> 40^\circ$) during more than half of the session. This induce self occlusions, drastic illumination changes and missing data. If we remove these samples, the error drops to $\approx 9.9^\circ$ (median of left eye). Here we stress that our method is indeed providing a mean to acquire head pose independence for gaze estimation.

In Fig. 4 we show the evolution of the estimated parameters as a function of time. There we observe that even under changes of the head pose, the estimated gaze follows closely the precomputed ground truth.

6. Conclusion

In this paper we have shown a novel approach to estimate gaze under free-head movements. The system combines depth and visual data, from a Microsoft Kinect sensor, to create a textured 3D mesh of the scene. We rigidly register a person-specific 3D face model, such that we are able to reliably track the head under challenging poses.

The estimated head-pose parameters are used to stabilize the 3D scene, and to generate eye images as if the camera was frontal to the head. This has the advantage of largely reducing the complexity of the gaze estimation problem. From the head-stabilized images, the gaze parameters are estimated from an appearance model and then transformed back according to the estimated head pose.

We have also shown a simple method to collect ground truth data using a Microsoft Kinect sensor. It uses a discriminative object that we can track in both the visual and depth domain. This method gives us an approximation of the visual gaze axis which we can use for training.

Experimental results show that our method successfully estimates the gaze direction under challenging head poses, and low resolution eye images. Even though our system does not achieve state-of-the-art results (which are as low as 1° in the HCI literature) we address a much less constrained problem, and the results show our approach is very promising. Moreover, this system is adequate to address tasks such as visual focus of attention, or studying gaze patterns for human behavior analysis.

Our future work will consist in a fine stabilization of the rectified eye-images, combined with a better representation of the image appearance. We expect this to increase the accuracy of the frontal gaze estimation thus improving the accuracy under free head movements.

Acknowledgements: The authors gratefully acknowledge the financial support from the Swiss National Science Foundation (Project: FNS-203, TRACOME) `www.snf.ch` and from the HUMAVIPS project, funded by the European Commission Seventh Framework Programme, Theme Cognitive Systems and Robotics, Grant agreement no. 247525.

References

[1] B. Amberg, R. Knothe, and T. Vetter. Expression invariant 3D face recognition with a Morphable Model. In *8th IEEE Int. Conf. on Automatic Face and Gesture Recognition*, Sept. 2008.

[2] G. Fanelli, T. Weise, J. Gall, and L. V. Gool. Real Time Head Pose Estimation from Consumer Depth Cameras. In *33rd Symposium of the German Association for Pattern Recognition (DAGM'11)*, Sept. 2011.

[3] L. Feng, Y. Sugano, O. Takahiro, and Y. Sato. Inferring Human Gaze from Appearance via Adaptive Linear Regression. In *ICCV: International Conference on Computer Vision*, Barcelona, Spain, 2011.

[4] L. Feng, O. Takahiro, Y. Sugano, and Y. Sato. A Head Pose-free Approach for Appearance-based Gaze Estimation. In *Proceedings of the British Machine Vision Conference*, 2011.

[5] D. W. Hansen and Q. Ji. In the eye of the beholder: a survey of models for eyes and gaze. *IEEE Trans. on Pat. Analysis and Machine Intelligence*, 32(3):478–500, Mar. 2010.

[6] C. D. Herrera, J. Kannala, and J. Heikkilä. Accurate and practical calibration of a depth and color camera pair. In *International Conference on Computer analysis of images and patterns - Vol II*, CAIP'11, pages 437–445, 2011.

[7] A. Kendon. Some functions of gaze-direction in social interaction. *Acta Psychologica*, 26:22–63, 1967.

[8] J. E. McGrath. *Groups: Interaction and Performance*. Prentice-Hall, 1984.

[9] E. Murphy-Chutorian and M. Trivedi. Head Pose Estimation in Computer Vision: A Survey. In *IEEE Trans. Pattern Anal. Machine Intell.*, 2008.

[10] E. Nguyen, C. Labit, and J.-M. Odobez. A ROI approach to hybrid image sequence coding. In *1st Int. Conf. on Image Processing*, volume 3, pages 245–249, Austin, Nov. 1994.

[11] B. Noris, J. Keller, and A. Billard. A wearable gaze tracking system for children in unconstrained environments. *Computer Vision and Image Understanding*, pages 1–27, 2010.

[12] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3D Face Model for Pose and Illumination Invariant Face Recognition. In *Proceedings of the 6th IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS) for Security, Safety and Monitoring in Smart Environments*, Genova, Italy, 2009. IEEE.

[13] Y. Sugano, Y. Matsushita, Y. Sato, and H. Koike. An incremental learning method for unconstrained gaze estimation. In *ECCV*, pages 656–667. Springer, 2008.

[14] T. Weise, S. Bouaziz, H. Li, and M. Pauly. Realtime performance-based facial animation. *ACM Trans. on Graphics (Proc. SIGGRAPH 2011)*, 30(4):1, July 2011.

[15] O. Williams, A. Blake, and R. Cipolla. Sparse and semi-supervised visual mapping with the S3GP. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, pages 230–237, 2006.

[16] Z. Zhang. Iterative Point Matching for Registration of Free-form Curves, 2004.