A Semi-Automated System for Accurate Gaze Coding in Natural Dyadic Interactions

Kenneth A. Funes-Mora, Laurent Nguyen, Daniel Gatica-Perez, Jean-Marc Odobez
Idiap Research Institute and École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
(kfunes,lnguyen,gatica,odobez)@idiap.ch

ABSTRACT

In this paper we propose a system capable of accurately coding gazing events in natural dyadic interactions. Contrary to previous works, our approach exploits the actual continuous gaze direction of a participant by leveraging on remote RGB-D sensors and a head pose-independent gaze estimation method. Our contributions are: i) we propose a system setup built from low-cost sensors and a technique to easily calibrate these sensors in a room with minimal assumptions; ii) we propose a method which, provided short manual annotations, can automatically detect gazing events in the rest of the sequence; iii) we demonstrate on substantially long, natural dyadic data that high accuracy can be obtained, showing the potential of our system. Our approach is non-invasive and does not require collaboration from the interactors. These characteristics are highly valuable in psychology and sociology research.

Categories and Subject Descriptors

I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding

Keywords

Gaze event detection, social interaction, RGB-D cameras

1. INTRODUCTION

In face-to-face interactions, gaze plays a crucial role as it is used to regulate the flow of communication, monitor feedback, reflect cognitive activity, express emotions, and communicate the nature of the interpersonal relationship [9]. Gazing patterns vary enormously according to the social setting: the personalities of the interactors, the topic of the conversation, or the other person's gazing patterns are all factors which might influence one's gaze. Related work in psychology has established the relationship between gaze behavior and social constructs such as dominance, cognitive ability and personality traits in social settings [9]. However, one of the important bottlenecks in social psychology research is the reliance on manual annotations, which makes the cost of conducting studies high and limits the size of data corpora; this is especially true for gaze.

ICMI'13, December 09–13, 2013, Sydney, Australia

ACM 978-1-4503-2129-7/13/12$15.00.
http://dx.doi.org/10.1145/2522848.2522884 .

Leveraging on related work in psychology, the social computing community has shown interest in using gaze as a nonverbal cue to analyze face-to-face interactions and automatically detect social variables such as dominance [7], addressee [8], or personality traits [10]. Due to the difficulty of automatically extracting gaze, most studies have used head pose as a substitute for visual focus of attention [7, 8]; while this approach has shown interesting results for the prediction of constructs in social interactions, it is inherently crude and unable to capture subtle gaze patterns. We observe a clear need from both the social computing and psychology communities for an accurate system to automatically code gazing events.

Within the computer vision field there have been important advances on the development of automatic gaze tracking methods [5]. However, most of these proposals focus on Human Computer Interfaces (HCI) applications and on people looking at screens, and often have restrictive assumptions. These systems are either invasive, highly expensive, or require user collaboration and/or constrained body and head movement. Less restrictive systems for human-human interaction analysis, capable of estimating head and gaze information, have been constrained to provide coarse measurements, e.g. discrete gaze pan [4].

Recent gaze estimation research aim at overcoming these limitations [2, 11]. Provided multimodal vision, such as standard and depth imaging, a method was proposed to remotely sense head pose and gaze, which obtains good performance under unrestricted head movements and low-resolution [2]. In this paper, we build upon this method and extend its application to an RGB-D camera pair, which jointly provides a 3D understanding of the scene and allows the monitoring of gaze events, like looking at another person. In this manner we aim at filling the current need for cheap and easy gaze coding.

In summary, this paper makes three contributions: i) we propose a system built from consumer RGB-D sensors appropriate for dyadic interactions, and we describe a technique to easily calibrate this camera pair in a room; ii) we propose a method which, provided short manual annotations, allows to remove the small gaze bias resulting from the use of a person-unspecific gaze model, and can automatically detect gazing events in the rest of the dyadic interaction; iii) we demonstrate that high gaze event detection accuracy can be achieved on long natural dyadic sequences. Our approach is general and could also be employed in an online manner.

2. PROPOSED SYSTEM

In this section we describe our proposed system, followed by the head pose and gaze estimation algorithm; we describe the gazing detection method, and finally the protocol needed to code gaze events on a long dyadic interaction.

2.1 System setup

For our proposed system we employed two RGB-D cameras (Kinects) positioned on a table and facing opposite directions, such that each camera monitors one person, as shown in Fig. 1.

System calibration is needed to allow for 3D scene understanding. This is achieved by first calibrating each RGB-D sensor once to obtain the intrinsic parameters of the RGB and depth cameras, together with the relative pose between them [6]. These calibration parameters allows us to interpret the RGB-D data as a textured 3D mesh in the camera coordinate system (**CCS**). In a second step, the 3D pose of each RGB-D camera needs to be estimated with respect to a fixed world coordinate system (**WCS**), which allows to construct a single textured 3D mesh from both RGB-D cameras and provide a rich representation helping to interpret the 3D geometry of the scene and interaction.

As the fields of view of the cameras do not overlap, we propose to leverage on the background walls to estimate the camera pose, and make the following assumptions which can usually be easily met in standard rooms: *a*) The wall planes are parallel *b*) Both cameras are at the same height (e.g. on a table) *c*) There is no rotation along the z axis of each camera (roll) *d*) The wall to wall distance and between camera's distance is measured in advance.

Given these assumptions, the camera pose estimation procedure works as follows: given a single depth frame we fit a plane to the wall depth pixel data obtained by distance thresholding. The camera's tilt and yaw are then obtained by aligning the wall's normal vector with the WCS's z axis, whereas the roll is assumed to be 0.

The position of each camera along the WCS's z axis is obtained from the camera-to-wall distance obtained from the fitted wall plane and the measured wall-to-wall distance. The camera position along the y axis is set to 0 (no difference in height) and the last parameter, the position along the x axis, is obtained from the measured camera-to-camera distance and the previously estimated pose parameters.

2.2 Head and gaze tracking

To detect gazing events it is necessary to track the head pose and gaze vectors for each person. To that end we build upon the approach of [2], which we summarize below.

Using a 3D Morphable Model (**3DMM**) [12] modeling facial variation, we first obtain a 3D mesh of the facial shape

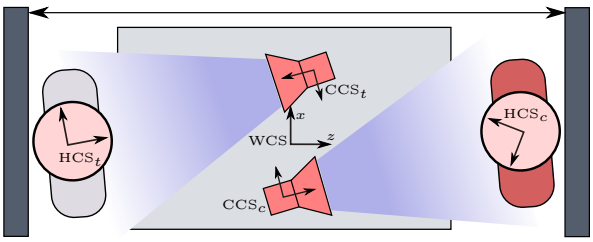


Figure 1: Top view of the system setup. Defining each of the 3D coordinate systems.

of each participant (template) by fitting the 3DMM to depth data. The fitting method is detailed in [1], which was extended to use facial landmarks to better constrain the fitting as in [2]. Given the personalized face template, the head pose and gaze direction is tracked frame by frame as follows.

Head pose. The Iterative closest point (ICP) method is used to estimate the head pose parameters $\mathbf{p} = \{\mathbf{R}, \mathbf{t}\}$, i.e. a 3D rotation and translation, by fitting the template's 3D pose to depth data. The frame to frame initialization is taken from the estimate on the previous frame. The overall initialization is based on face detection in the RGB video.

Gaze estimation. Given the estimated head pose we rectify the eye image appearance, due to head pose variation, by rendering the textured 3D mesh of the scene using the inverse of the head pose parameters $\mathbf{p}^{-1} = \{\mathbf{R}^\top, -\mathbf{R}^\top\mathbf{t}\}$. Due to the semantics inherited from the 3DMM, we know an approximate location of the eyeball center "$\mathbf{o}$", which we use to crop the now frontal looking eye images. These images are used to estimate the eye-in-head gaze direction.

The gaze estimation approach here used was initially proposed by Lu et al. [11] for frontal head pose gaze estimation. Assuming that a set of eye images $\{\mathbf{I}_i\}$, with associated gaze directions $\{\mathbf{g}_i\}$ is available they formulated the problem as the sparse reconstruction of a test eye image $\hat{\mathbf{I}}$ from $\{\mathbf{I}_i\}$.

The sparse weights vector $\{\hat{w}_i\}$, which best reconstructs the test image, is used to linearly combine the gaze parameters into the test image gaze direction as $\hat{\mathbf{g}} = \sum_i \hat{w}_i \mathbf{g}_i$.

Similar to [2], the gaze parameters $\mathbf{g} \in \mathbb{R}^2$ correspond to the gaze angular yaw and elevation, defined in the head coordinate system (**HCS**). We refer the reader to [2] for further details which are out of the scope of this discussion.

If necessary, the gaze direction $\hat{\mathbf{g}}$ can be referred to the CCS. This transformation is given by the head pose parameters $\mathbf{p}$. This procedure makes the gaze direction to be represented by a 3D vector pointing out from the point $\mathbf{o}_C = \mathbf{R}\mathbf{o} + \mathbf{t}$. Notice that each eye is handled separately.

2.3 Gaze event detection

To infer whether a person is looking at a visual target, two main elements are needed: the target position and the 3D gaze direction, both referred to the same coordinate system. Here the visual target $\mathbf{y} \in \mathbb{R}^3$ is defined as a fixed point between the eyes of the other person, referred to his/hers HCS. The gaze direction and the head pose is estimated as described in Section 2.2. All quantities can be referred to the WCS or any HCS using the system geometry described in Section 2.1 and shown in Fig. 1.

The gazing event, i.e., when a participant looks at the other person, occurs when his/hers 3D gaze vector intersects the visual target. In practice this intersection is difficult to detect due to low-resolution imaging, depth noise, head pose tracking jitter, and the lack of a gaze appearance model $\mathcal{A} := \{(\mathbf{I}, \mathbf{g})_i\}$ specific to the given subject.

In addition, differences in appearance and cropping positioning of the eye images, between the test user and those users within $\mathcal{A}$, can introduce systematic deviations between the estimated gaze and its actual values. We alleviate this deviation by introducing a systematic gaze bias $\mathbf{b}_c$.

We can then proceed to define the gazing event decision function. Let $\Phi(\mathbf{g}) : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ be a transformation of the gaze angular parameters to a equivalent unitary 3D vector defined in the HCS. Given the head poses of the current and target person: $\mathbf{p}_c := \{\mathbf{R}_c, \mathbf{t}_c\}$ and $\mathbf{p}_t := \{\mathbf{R}_t, \mathbf{t}_t\}$ respec-

(a) (b) (c)

Figure 2: Annotations: (a) Facial landmarks; (b) Gazing events; (c) Not gazing events: shown for comparison to (b).

tively (referred to the WCS) we define the gaze reference vector, in the current person's HCS, as the vector which points from the eyeball to the target as $\mathbf{v_y} = \hat{\mathbf{v}}_\mathbf{y}/||\hat{\mathbf{v}_\mathbf{y}}||$ where:

$$\hat{\mathbf{v}}_\mathbf{y} = \left(\mathbf{R}_c^\top \left(\mathbf{R}_t\mathbf{y} + \mathbf{t}_t\right) - \mathbf{R}_c^\top \mathbf{t}_c\right) - \mathbf{o}. \quad (1)$$

Finally the gazing decision function is:

$$\arccos\left(\Phi(\mathbf{g} + \mathbf{b}_c) \cdot \mathbf{v_y}\right) < \tau, \quad (2)$$

where τ is the gazing angular threshold. We discuss the estimation of τ in Section 3.2.

Bias estimation. Given annotated samples of gaze events, we can compute the gaze angular error (in $\mathbb{R}^2$) as the difference between the gaze reference (as yaw/elevation) and the estimated $\mathbf{g}$. We then set $\mathbf{b}_c$ as the geometric median of these differences, i.e., the point which minimizes the sum of distances to all other points labeled as gaze events.

2.4 Semi-automated gaze coding

We now summarize the needed steps for a user of our system to code gazing events in a long dyadic interaction.

1. Set the system according to the assumptions, and measure for calibration purposes the wall to wall and camera to camera distances (Section 2.1).
2. For each participant, place facial landmarks (see Fig. 2a) in one or two frames from the entire sequence to build the facial shape template (Section 2.2). Note that a person model needs to be built only once, and can be re-used if the person appears in several sequences.
3. To further improve results, code gazing events from a small sequence of the interaction (see samples in Fig. 2b) to estimate the gaze angular bias (Section 2.3).

3. EXPERIMENTS

3.1 Data

We collected the training samples for gaze estimation, i.e. $\{(\mathbf{I}, \mathbf{g})_i\}$, from 5 people different from the participants involved in our evaluations. The aggregation of samples from different users have proven useful when it is not possible to collect samples for the given test subject [3]. Five natural dyadic interactions between an interviewer and a job candidate were recorded using two RGB-D sensors. In order to train and evaluate our gaze detection system, one person manually annotated gazing events for both protagonists of the interaction. Gazing events were defined as events when the person of interest was looking at the other protagonist. As manual annotations of gaze are time-intensive, only the first five minutes of the interactions were annotated. The annotations were done by noting the starting time and end of gazing events; in order to ensure that the timings were accurate, we generated subtitle files and played back the video with the subtitles in VLC, and adjusted the timing such that they were accurate. In the following experiments the interviewer will be referred as participant 6.

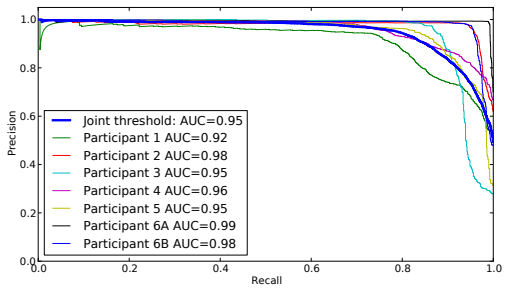


Figure 4: Precision recall curve for gazing detection for different participants. AUC=area under the curve.

Table 1: Gazing detection accuracy with (B) or without (NB) bias correction.

	Participant							
Method	1	2	3	4	5	6A	6B	Mean
Head NB	54.7	55.4	53.8	74.8	46.3	72.9	79.2	62.4
Head B	62.5	70.9	49.0	62.3	46.3	77.4	81.3	64.2
Gaze NB	83.3	92.4	84.2	84.3	75.3	85.3	84.6	84.2
Gaze B	81.1	91.4	85.5	81.3	79.8	92.3	90.6	86.0

3.2 Gazing detection threshold setting

Fig. 4 shows the precision-recall (PR) curves for the gaze detection event evaluated at frame level over an annotated extract ($\sim$ 2 min) per participant. We computed a bias for each case using the approach described in Section 2.3. Each point on the curves is obtained by setting a different threshold τ and estimating the precision recall values on the same data.

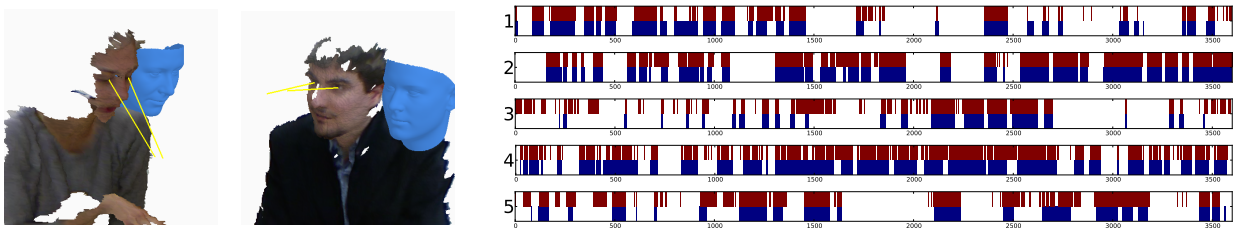
Fig. 4 also shows the joint PR curve, obtained using the same τ for all participants and computing the average precision and recall. As can be seen, this curve follows relatively closely the individual PR curves, showing the consistency of performance with respect to the threshold. To set the threshold for the evaluation below, we used the equal error rate (EER) $\tau \approx 12^\circ$ value.

3.3 Gazing detection accuracy

In Table 1 we show a comparison of gaze event detection methods at frame level. We include the case of using the head pose only as a proxy for gaze event detection. In both cases, with gaze estimation and head only, we evaluated with and without bias. Adding bias for the head pose indirectly helps to select the most observed head pose when gazing. When applicable, the bias is obtained from one minute of the interaction. Also, τ was set as described in Section 3.2. The evaluation is done on the remaining 4 minutes.

From these results we can conclude the following : i) The head pose alone is helpful but not sufficient for gazing events detection; ii) Introducing gaze information highly improves the accuracy with respect to head pose; iii) The bias helps to improve the results in most cases, but not necessarily.

The last point indicates the system can be accurate even without coding a small sequence of the interaction.



(a) (b)

Figure 3: Qualitative results: (a) 3D rendering of an interaction; (b) Time plots of 2 minutes of gazing events. We show manually annotated events (blue) vs. estimated gazing events (red) for participants 1 to 5.

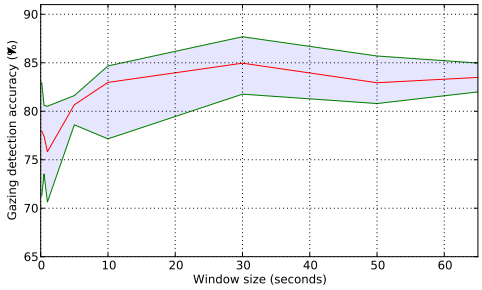


Figure 5: Gazing detection accuracy in function of the training window size. The minimum and maximum accuracy obtained when training using different sections of the video are also displayed.

3.4 Training window selection

Here we study the effect of gaze events samples selection (manual annotation) over the system accuracy on the detection for the rest of the interaction detection. To this end, we defined training windows of different sizes and positions and generated the plot shown in Fig. 5. This plot is averaged over the different participants.

As we can see, accuracy does increase as the window size is larger. However, it quickly saturates to a small window size, showing that manual annotations can be as short as 20 seconds. The variations in accuracy means that it is important to select a "good" window. Generally speaking, this can be achieved by including gazing events in varying circumstances (e.g. head poses).

3.5 Qualitative results

Fig. 3a shows an example of gaze result on a sample frame of interaction. Qualitatively we see the gain of including gaze information over head pose alone (esp. for tilt). In Fig. 3b we show a comparison of the manually annotated gaze events vs. the automatic results produced at frame level (30fps). We see that our method properly follows the gazing behavior. Notice that gaze event estimation is currently conducted frame by frame, and no post filtering is applied. There is thus important room for improvement, e.g. by exploiting also the gaze dynamics, or multimodal information such as the participant's speaking status.

4. CONCLUSION

We presented a semi-automated system to detect gazing events at the frame level in dyadic interactions. Our system exploits the 3D understanding of the interaction by tracking the 3D head poses and continuous gaze vectors of the participants. The proposed system has the advantages of being easy to calibrate and non-intrusive for the participants. This makes it adequate for studies involving dyadic interactions. We have conducted evaluations on natural interactions, and reported high accuracy in the detection of gazing events. We showed that including the eye gaze information improves the detection accuracy over using head pose only, as done in most previous works. We emphasize that our system requires minimal effort from the user to code gazing events in an interaction. We have demonstrated that a manual coding of 20 seconds of the interactions is sufficient to achieve 85% of accuracy on average.

For future work, we will address the detection of gazing events in small group interactions. This requires an extension of this system to handle multiple visual targets, which would also be valuable for other applications.

Acknowledgments The authors gratefully acknowledge the financial support from the Swiss National Science Foundation (Projects: FNS-203, TRACOME and FNS-194, SONVB) `www.snf.ch`. We also thank Denise Frauendorfer (Université de Neuchâtel) for the support with data collection.

5. REFERENCES

[1] B. Amberg, R. Knothe, and T. Vetter. Expression invariant 3D face recognition with a Morphable Model. In *2008 8th IEEE International Conference on Automatic Face and Gesture Recognition*, pages 1–6. IEEE, Sept. 2008.

[2] K. A. Funes Mora and J.-M. Odobez. Gaze Estimation From Multimodal Kinect Data. In *Computer Vision and Pattern Recognition Workshops*, pages 25–30, June 2012.

[3] K. A. Funes Mora and J.-M. Odobez. Person Independent 3D Gaze Estimation From Remote RGB-D Cameras. In *International Conference on Image Processing*, Sept. 2013.

[4] S. Gorga and K. Otsuka. Conversation scene analysis based on dynamic Bayesian network and image-based gaze detection. In *ICMI*, 2010.

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

[6] D. Herrera C., J. Kannala, and J. Heikkilä. Joint Depth and Color Camera Calibration with Distortion Correction. *TPAMI*, 34(10):2058–2064, 2012.

[7] H. Hung, D. Jayagopi, S. Ba, J. Odobez, and D. Gatica-Perez. Investigating automatic dominance estimation in groups from visual attention and speaking activity. In *Int. Conf. on Multimodal Interfaces*, 2008.

[8] D. Jayagopi, D. Sanchez-Cortes, K. Otsuka, J. Yamato, and D. Gatica-Perez. Linking speaking and looking behavior patterns with group composition, perception, and performance. In *ICMI*, page 433, New York, 2012.

[9] M. L. Knapp and J. A. Hall. *Nonverbal communication in human interaction*. Wadsworth, Cengage Learning, 7 edition, 2009.

[10] B. Lepri, R. Subramanian, and K. Kalimeri. Employing social gaze and speaking activity for automatic determination of the extraversion trait. In *ICMI*, 2010.

[11] F. Lu, Y. Sugano, O. Takahiro, and Y. Sato. Inferring Human Gaze from Appearance via Adaptive Linear Regression. In *ICCV*, Barcelona, Spain, 2011.

[12] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3D Face Model for Pose and Illumination Invariant Face Recognition. In *AVSS*, Genova, Italy, 2009.