Noname manuscript No.
(will be inserted by the editor)

Gaze Estimation in the 3D Space Using RGB-D sensors

Towards Head-Pose And User Invariance

Kenneth A. Funes-Mora · Jean-Marc Odobez

Received: date / Accepted: date

Abstract We address the problem of 3D gaze estimation within a 3D environment from remote sensors,which is highly valuable for applications in human-human and human-robot interactions. To the contrary of most previous works, which are limited to screen gazing applications, we propose to leverage the depth data of RGB-D cameras to perform an accurate head pose tracking, acquire head pose invariance through a 3D rectification process that renders head pose dependent eye images into a canonical viewpoint, and computes the line-of-sight in 3D space. To address the low resolution issue of the eye image resulting from the use of remote sensors, we rely on the appearance based gaze estimation paradigm, which has demonstrated robustness against this factor. In this context, we do a comparative study of recent appearance based strategies within our framework, study the generalization of these methods to unseen individual, and propose a cross-user eye image alignment technique relying on the direct registration of gaze-synchronized eye images. We demonstrate the validity of our approach through extensive gaze estimation experiments on a public dataset as well as a gaze coding task applied to natural job interviews.

Keywords Gaze estimation · appearance based methods · RGB-D cameras · head-pose invariance · person invariance

Kenneth A. Funes-Mora
Idiap Research Institute, Switzerland
École Polytechnique Fédéral de Lausanne, Switzerland
Tel.: +41 277 21 77 01
E-mail: kenneth.funes@idiap.ch

Jean-Marc Odobez
Idiap Research Institute, Switzerland
École Polytechnique Fédéral de Lausanne, Switzerland
Tel.: +41 27 721 77 26
E-mail: jean-marc.odobez@idiap.ch

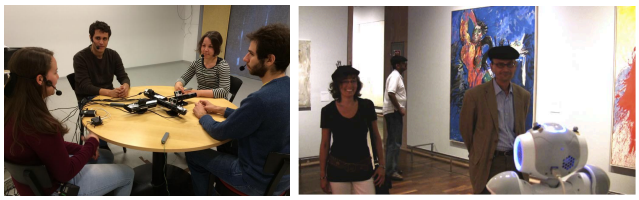

Fig. 1 Example of applications requiring 3D gaze estimation. Left: studying Human-Human Interactions in group interviews (Oertel et al, 2014). Right: Human-Robot Interactions involving groups of people.

1 Introduction

1.1 Motivations

The automatic estimation of gaze has an utmost importance for a wide range of fields of study and applications. Indeed, gaze is acknowledged as one of the most important non-verbal communication cues. It is known to be highly involved in the regulation of the conversation flow, especially within groups, and to convey information about a subject's intentions, inner states, and even psychological traits. Therefore, automatic gaze estimation can greatly help further and larger scale studies in psychology and sociology, which have predominantly relied on manual annotations or crude approximations for gaze. More generally, as a display of user attention, gaze is also a valuable cue in the development of intuitive human computer interfaces (HCI) or in the design of natural human robot interactions (HRI). Other applications include marketing, entertainment, assistive driving and navigation, assistance for users with limited body motion, etc.

In the past, significant efforts have been devoted to the design of automatic gaze estimation solutions, leading to methods which differ according to their sens-

ing technique and principles: from the highly intrusive electro-oculography to the more flexible video-oculography, i.e., gaze tracking relying on video input (Hansen and Ji, 2010). Even though the latter has higher potential for practical applications, it needs to address important challenges. In particular, the eye appearance varies depending on the user, head pose, illumination conditions, image resolution and contrast, eyelids shape and movements, specular reflections, motion blur, self occlusions, etc.

To overcome some of these challenges, many gaze estimation systems, in particular those readily available in the market, rely on specialized hardware like head mounted cameras and/or infrared (IR) setups. The advantage of the former is the capture of standardized eye images, i.e., with a single scale and viewpoint. Nevertheless, for many applications, head mounted sensors are still considered as intrusive. Infrared setups profit from the bright/dark pupil effect and the reflections of calibrated IR light sources in the cornea. However, it requires high resolution imaging and potentially expensive IR hardware. Thus, natural light based methods using remote sensors still remain the best candidates in terms of hardware availability, cost and applications. Yet, many of the aforementioned challenges are far from being solved when using consumer cameras.

The advent of inexpensive depth sensors may however help to address these challenges. Indeed, in the recent past, such sensors have allowed researchers to handle problems known to be highly challenging when based on standard vision alone (Murphy-Chutorian and Trivedi, 2008), such as body pose estimation (Shotton et al, 2011) or facial expressions recognition (Weise et al, 2011). Through depth (D) maps, these sensors provide explicit and reliable measurements of the scene's shape, as opposed to the implicit shape information embedded within the RGB data. Notice it is still difficult and costly to infer shape information from the visual domain (RGB) alone (Barron and Malik, 2013).

Therefore, depth sensing enables the use of shape information in further processing stages. In particular, depth data has been shown to be valuable for accurate head pose estimation (Fanelli et al, 2011; Weise et al, 2011), a necessary step prior to determining the gaze direction. On the other hand, gaze itself requires standard vision measurements to determine the eye orientation from the eye image, and the most important challenges to address are the eye appearance variabilities due to head pose, users, and the low eye image resolution when considering applications that do not restrict the mobility of users. In this regard, Fig. 2 illustrates the eye images obtained using a Kinect as compared to the higher resolution images considered in other works.

Fig. 2 Sample eye images taken from a) Martinez et al (2012) b) Noris et al (2010) c) Lu et al (2011) d) Schneider et al (2014). e) Samples Kinect-based eye images from the EYEDIAP database (Funes Mora et al, 2014), taken at $\approx$ 1m; Notice the much poorer resolution and contrast.

1.2 Approach and contributions

We address the problem of appearance based gaze estimation under varying head poses and users using remote RGB-D cameras. A main contribution is to propose a methodology which profits from consumer RGB-D sensors to rectify the eye appearance into a canonical head pose viewpoint. This relies on the depth information for both performing an accurate tracking of the head pose and doing a depth-driven warping of the eye texture as a function of the estimated head-pose; either the raw depth information, delivered by the sensor, or the depth of the eyeball surface induced by the fitted 3D Morphable Model (3DMM).

The main benefit of the rectification framework is to bring head-pose invariance into existent appearance-based gaze estimation methods. In this context, we evaluate the performance of recent models, for both user specific and user independent cases. In particular, to address the problem of eye region cropping and alignment, which is a crucial step to avoid biases in the gaze estimation, we propose a new method using a synchronized alignment approach.

Extensive evaluations on a public database featuring 16 users performing two types of tasks (looking at a screen and, more challenging, looking a target in the 3D space) under fixed or mobile head poses show promising results of the proposed methodology. The validity of our approach is further demonstrated in a gaze coding task applied to job interview interaction data.

This paper is structured as follows. Related works are discussed in Section 2. The head-pose invariant appearance based gaze estimation framework is described in Section 3. Gaze appearance methods, suitable for this context, are briefly described in Section 4. Extensions to acquire user invariance are described in Section 5. Gaze estimation data and experimental protocols are presented in Section 6, followed by results in Section 7. Section 8 presents experiments on the automatic gaze coding task. Section 9 discusses limitations and future work. Finally, Section 10 concludes this work.

2 Related work

The automatic estimation of gaze has been well investigated for over three decades, as well described by Hansen and Ji (2010). Two main strategies are identified: geometric and appearance based methods.

Geometric based methods. They rely on the detection of local features which are mapped to gaze parameters. Most methods require a calibration session to collect gaze annotated samples. These are used to determine user specific parameters describing the eyeball geometry or a direct mapping to the point of regard.

The most accurate techniques in this category rely on IR illumination and sensing. This leads to the bright and dark pupil effect and the generation of specular reflections in the corneal surface, known as glints. Then the gaze direction can be inferred from the pupil center and the corneal reflections locations (Guestrin and Eizenman, 2006). These methods achieve head pose invariance using multiple IR light sources. However, specialized and costly IR hardware is needed.

Under natural light conditions, many proposals also leverage local eye features to build geometric models of the eyes. Features such as the iris center (Timm and Barth, 2011; Valenti and Gevers, 2012), an ellipse fitted to the pupil/iris (Li et al, 2005), or even complex shapes incorporating the eyelids (Yuille et al, 1992) or the full eye region (Moriyama and Cohn, 2004) could be used. As an example, the commercial system faceshift[1], designed for facial motion capture from consumer RGB-D sensors, makes use of this principle for eye tracking.

Among prior works, Ishikawa et al (2004) rely on iris ellipse fitting, from which the eyeball geometric parameters are found through a careful calibration protocol. This model is then used to compute gaze from ellipse fitting at test time. Yamazoe et al (2008) proposed a similar strategy with a reduced calibration session. Ellipse fitting was also used, but obtained from a prior segmentation of the eye region based on thresholding.

Recent methods apply a similar strategy to RGB-D data. Jianfeng and Shigang (2014) infer gaze based on iris center localization and the Microsoft Kinect SDK's head pose tracker. The eyeball center is refined from a calibration session, whereas the rest of eyeball parameters are fixed. Xiong et al (2014) used the same sensor, but relied on ellipse fitting and facial landmarks for 3D head pose tracking as well as to build a person specific facial landmarks position model. Their calibration method infers additional eyeball parameters. However, in both cases, the Kinect was configured for the highest RGB resolution of 1280×960 to allow local features tracking. In addition, the evaluated range of gaze directions was small and head pose variations were minimal.

Nevertheless, an important limitation of the previous methods is the need to detect local features, which require high resolution and high contrast images.

Appearance based methods. By modeling a direct mapping from the entire eye image to gaze parameters (Baluja and Pomerleau, 1994; Tan et al, 2002; Funes Mora and Odobez, 2012; Lu et al, 2011; Martinez et al, 2012; Sugano et al, 2008; Noris et al, 2010), these approaches avoid the cumbersome local features tracking, providing potential for low-resolution gaze sensing.

As a pioneering work, Baluja and Pomerleau (1994) rely on a neural network but require thousands of training samples to reduce the gaze estimation error. Tan et al (2002) use linear interpolation from a local gaze appearance manifold, which also requires a large training set. Williams et al (2006) present a semi-supervised sparse Gaussian Process Regression (S^3GPR) method to reduce the required training samples by profiting from weakly labeled samples.

In a similar vein, Sugano et al (2008) propose to exploit user-computer interactions traces as training data within a local linear interpolation method relying on the prior clustering of samples based on head pose. To avoid the local manifold selection prior to linear interpolation, Lu et al (2011) propose to use sparsity while interpolating from all samples. They report high accuracy, even for low-resolution images, but these images were artificially created from the same session and the method require a fixed head pose using a chin-rest. The same authors later integrated blinks detection and an improved subpixel alignment approach within their framework (Lu et al, 2014b). Their overall method allows to estimate gaze under low-resolution and slight head motion ($\approx$ static), thus not requiring a chin rest.

Noris et al (2010) propose to train a Support Vector Regression (SVR) model using the stacked eye image pixels as the appearance feature vector, which is first preprocessed to handle illumination variations. Alternatively, Martinez et al (2012) propose to use multi-level Histogram of Oriented Gradients (mHoG) (Dalal and Triggs, 2005) as appearance features to train a SVR or Relevance Vector Regression (RVR) model. The advantage is that HoG can better cope with illumination variations, in contrast to the intensity based features. Nevertheless, in both cases, the methods are proposed for head-mounted cameras, i.e., a single viewpoint.

Invariance. In spite of their robustness to image resolution, appearance based methods suffer from generalization problems. Few of the previous works address head pose invariance and all these approaches are trained and tested on the same individual.

[1] `www.faceshift.com`

To address head pose variations, Lu et al (2014a) propose a GPR-based correction of the gaze parameters bias caused by the head pose. Alternatively, they propose to use a single pose gaze appearance model and a few samples from different head poses to warp the known set as seen from the test viewpoint (head-pose) (Lu et al, 2012). Altogether, however, these methods still require additional training data and complex models to capture head-pose related appearance variations.

In another direction Funes Mora and Odobez (2012) propose to use depth data to rectify the eye appearance into a canonical head viewpoint, an approach which was later used by Egger et al (2014) relying on a 3D face model fitted to the 2D image, rather than depth data.

The person invariance problem for appearance based gaze estimation (also known as "calibration-free") is gaining higher interest. Noris et al (2008) train a GPR model from eye image samples retrieved from a large group of people. However, this method is designed for head-mounted sensors. Funes Mora and Odobez (2013) propose to adopt an unsupervised model selection based on the assumption that the sparse reconstruction of a test sample will select subjects within the given database through the reconstruction weights. Schneider et al (2014) develop a dimensionality reduction method which maximizes the intra-class distance while minimizing the inter-class distance of gaze synchronized samples. Very recently, Sugano et al (2014) propose to train random forests for regressing the gaze parameters from eye appearance and head pose estimates. To increase the training set they use a multi-view camera array for data collection, allowing to synthesize pose-dependent samples using 3D multi-view reconstruction. By including different subjects the model is also trained for person invariance. Although promising, evaluations were conducted assuming a perfect eye detection, i.e., by annotating manually the eye corners.

Good eye image localization (or alignment) is an important step to achieve high performance, since it directly impacts the extraction of the eye feature vector further used in the regression methods. This problem, however, has not received much attention. Mostly because, when working with user and single session dependent models (or with little pose variations), a single cropping is assumed which usually remains consistent for all data points. In many cases, this step is assumed to be done manually (Martinez et al, 2012; Lu et al, 2011; Sugano et al, 2014). Alternatively, automatic eye corner detection methods can be achieved using the Omron software (Schneider et al, 2014), but normally requires high resolution images.

Contributions. We aim at addressing the problem of remote gaze estimation within the 3D space, accommodating large user 3D movements and head poses without requiring further training and while handling a large range of gaze directions. Such a formulation allows the natural application of gaze techniques to more diverse HHI (human human interactions) or HRI scenarios besides the traditional screen gazing case. To our knowledge, no previous works relying on the appearance-based paradigm have acquired these characteristics, either methodologically or empirically, and very few methods have addressed simultaneously head pose and user invariance. Note that while some previous methodologies devoted to the 2D screen-gazing case could be modified to handle the 3D case, they would in general be modified to at least require head pose estimation and a framework to handle variations in head pose. This is what we provide in this paper,

To this end, we develop further our head pose invariant framework designed for appearance-based methods using RGB-D cameras (Funes Mora and Odobez, 2012) that relies on the 3D rectification of the eye region appearance into a canonical viewpoint to address pose invariance. We study two rectification methods, further propose an alignment method to perform a finer cropping of the eye image, and show that this methodology is applicable with several recent state-of-the-art user independent appearance based methods. Promising results are obtained despite the low resolution of the input eye images, large range of gaze directions and significant pose variations.

3 Head pose invariant gaze estimation

In this section we describe our 3D rectification methodology for head pose invariant appearance based gaze estimation. We first introduce the overall approach, and then detail the different steps involved in the rectification process.

3.1 Approach overview

The main principle of our approach is to rectify the eye images into a canonical (frontal) head viewpoint and scale regardless of the actual head pose by exploiting the calibrated RGB-D input data, and then estimate the gaze in this canonical view.

The different steps involved in this process are depicted in Fig. 3. First, to obtain accurate head pose, we assume that a user specific 3D model is available. Currently, this model is learned in an offline step. Then, in the online phase, the proposed method consists of the following steps:

1. At each time step t, the 3D head pose $\mathbf{p}_t$ is estimated.

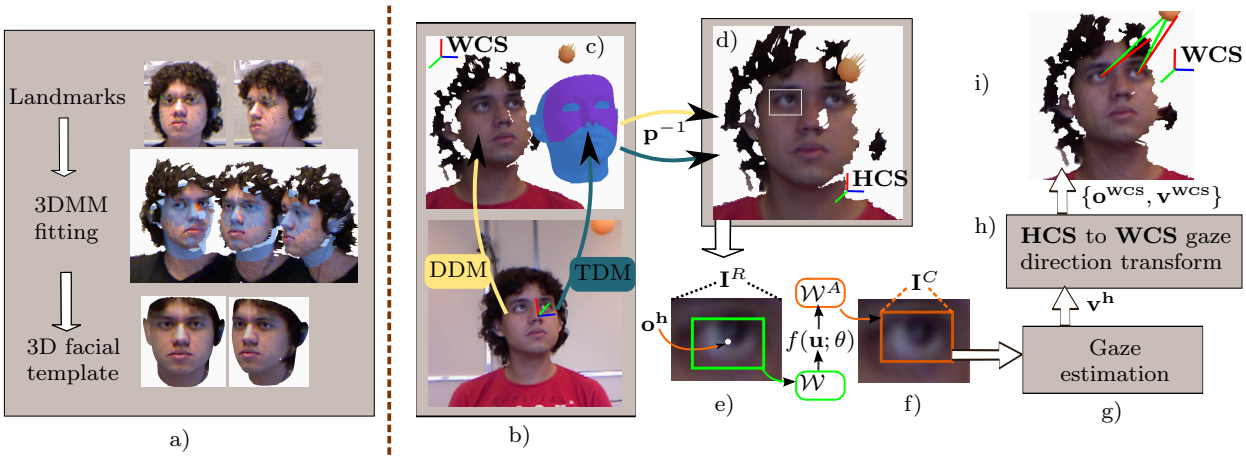

Fig. 3 Proposed method pipeline. a) Offline step. From multiple 3D face instances, the 3DMM is fit to obtain a person specific 3D model. b-i) Online steps. b) The 3DMM fitted face model is registered at each instant to the depth data of the RGB to obtain the head pose (the region used for tracking is rendered in purple in the 3DMM mesh in c). c) a 3D textured mesh is obtained by binding the RGB image either to the depth D channel of the sensor (shown Data Driven Mesh, *DDM*), or to the 3D facial template (template-driven mesh, *TDM*; note: only the template is shown). d) the textured mesh is rendered in a frontal pose by rotating it using the inverse head pose parameters, for which an eye image region $\mathbf{I}^R$ can be obtained. e-f) as a predefined region $\mathcal{W}$ around the eyeball center $\mathbf{o}^\mathbf{h}$ may not consistently crop the same eye part across users, an alignment warping f learned for each user is applied to $\mathcal{W}$ and defines the region $\mathcal{W}^A$ where the image should be cropped. g-i) the gaze $\mathbf{v}^\mathbf{h}$ in the head coordinate system **HCS** is estimated from the cropped image $\mathbf{I}^C$, and then transformed back in the **WCS** to obtain the line of sight (green line, estimated LoS; red line, ground truth).

2. The face region is rectified into a frontal view from the input RGB-D data and the estimated head pose, leading to a rendered image $\mathbf{I}^R$ for each eye. An eye alignment step is then applied in order to crop the eye region $\mathbf{I}^C$.
3. The gaze direction $\mathbf{v}^\mathbf{h}$ in the head coordinate system is estimated from $\mathbf{I}^C$. It is mapped back into the world coordinate system (**WCS**) using the pose $\mathbf{p}_t$, and used along with the eyeball center $\mathbf{o}^{\mathbf{WCS}}$ to define the gaze line of sight (LoS).

In the following, we describe the aforementioned steps in detail, starting by the offline step of building the user specific 3D facial template.

3.2 3D facial template creation

We propose to create a user-specific 3D facial template by fitting a 3D Morphable Model (3DMM) to input data. Such 3DMMs present the advantage of being able to generate a large variety of possible face shapes (i.e., person specific face shapes) using a relatively small set of coefficients. These coefficients can be found for a given subject from few face instances.

More precisely, a 3DMM shape can deform according to:

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

where $\mathbf{x} \in \mathbb{R}^{3N_v}$ (the model instance) denotes a set of 3D vertices coordinates $\{\mathbf{x}_i, i = 1, \ldots, N_v\}$ stacked as a large column vector, $\mu \in \mathbb{R}^{3N_v}$ is the mean shape and $\mathbf{M} \in \mathbb{R}^{3N_v \times N_\mathbf{M}}$ is formed from the $N_\mathbf{M}$ shape basis vectors. The model is therefore parametrized by the vector $\alpha \in \mathbb{R}^{N_\mathbf{M}}$.

Since the mesh topology is kept fixed, semantic information (such as eyeball location or eye surface) can be defined in the 3DMM topology and inherited by its person specific instances.

To learn the person specific 3D mesh, we find the model instance that best fits a set of J samples (RGB-D images) of the subject, by iteratively solving the following optimization problem:

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \left(\lambda E_s(\mathbf{X}) + \sum_{j=1}^{J} E_d^j(\mathbf{X}) + \gamma E_l^j(\mathbf{X}) \right), \tag{2}$$

where the parameters $\mathbf{X} := \{\alpha, \mathbf{R}_1, \mathbf{t}_1, \ldots, \mathbf{R}_J, \mathbf{t}_J\}$ to optimize are the coefficients α of the person 3DMM, and those of a rigid transformation (defined by a rotation $\mathbf{R}_j$ and translation $\mathbf{t}_j$) for each image instance j. The different cost terms are defined as follows:

$$E_d^j(\mathbf{X}) := \sum_{i=1}^{N_v} w_i \|\mathbf{R}_j(\mu_i + \mathbf{M}_i\alpha) + \mathbf{t}_j - \mathbf{u}_i\|^2,$$
$$E_l^j(\mathbf{X}) := \sum_{i \in L} \|\mathbf{R}_j(\mu_i + \mathbf{M}_i\alpha) + \mathbf{t}_j - \mathbf{l}_i\|^2,$$
$$E_s(\mathbf{X}) := \|\alpha\|^2, \tag{3}$$

where μ_i and $\mathbf{M}_i$ represent the 3 rows corresponding to the vertex i in μ and $\mathbf{M}$. The data term E_d represents the cumulative distance of each deformed and

rigidly transformed vertex i of the 3DMM to its closest point in the data, represented by $\mathbf{u}_i$. The per-vertex weight w_i is intended to provide robustness against outliers. It is re-estimated at each iteration k (see below), and defined inversely proportional to the euclidean distance between the current position of the mesh vertex i (given the current parameters $\mathbf{X}^k$) and its correspondence point. It is also set to zero if the angle between the surface normals in the current fit and at the correspondance $\mathbf{u}_i$ is above a threshold.

The term E_l is similar to the E_d cost, but applies to a set of landmarks points (which form a subset L of the 3DMM vertices) whose position $\mathbf{l}_i$ is assumed to be annotated in the data. This term fosters a semantic fitting of the 3DMM (eye corners, eyebrows, mouth corners, etc.) which, due to depth noise in the data, could be otherwise poorly localized. Finally, the regularization term E_s fosters the estimation of small values for α. This term is weighted by the *stiffness* parameter λ, controlling how much the instance mesh can deform.

The formulation in Eq. 2 is an extension of the optimization proposed in Amberg et al (2008), to which we have included the landmarks term and the possibility for multiple data samples, to compensate for noise (Kinect data are more noisy than laser scans) and partial depth observations. To find the optimal parameters $\hat{\mathbf{X}}$, we proceed iteratively as follows:

- Initialize $\mathbf{X}$ as $\mathbf{X}^0$. Then, for each stiffness value $\lambda^n \in \{\lambda^1, \ldots, \lambda^N\}, \lambda^n > \lambda^{n+1}$, do:
  - Until $\|\mathbf{X}^k - \mathbf{X}^{k-1}\| < \epsilon$:
    - find correspondences $\mathbf{u}_i$ in the target surface for each point i of the current 3DMM instance.
    - compute the weight w_i.
    - determine $\mathbf{X}^k$ by solving Eq. 2 using λ^n.

The initialization ($\mathbf{X}^0 := \{\alpha^0, \mathbf{R}_1^0, \mathbf{t}_1^0, \ldots, \mathbf{R}_J^0, \mathbf{t}_J^0\}$) is given by the mean face shape ($\alpha^0 = \mathbf{0}$) and its -per image sample j- rigid transformation ($\mathbf{R}_1^0, \mathbf{t}_1^j$) parameters minimizing the landmarks term E_l^j alone assuming $\alpha = \alpha^0 = \mathbf{0}$, i.e., the rigid transform that best fit the mean face shape to the annotated landmarks.

This algorithm systematically reduce the stiffness value allowing for larger deformations as the correspondences are more accurate. This is a common strategy in non-rigid Iterative Closest Points (ICP) methods (Amberg et al, 2007).

3.3 Head pose and eyes tracking

In the online phase we can now track the head pose by registering the person-specific template mesh to depth data. To this end, we use the ICP algorithm with point-to-plane constraints and find, for frame t, the pose parameters $\mathbf{p}_t = \{\mathbf{R}_t, \mathbf{t}_t\}$ minimizing the cost:

$$E(\mathbf{R}_t, \mathbf{t}_t) = \sum_{i \in U_H} w_i \left(\mathbf{n}_i^\top \left(\mathbf{R}_t \mathbf{v}_i + \mathbf{t}_t - \mathbf{u}_i\right)\right)^2 \quad (4)$$

in which $\mathbf{n}_i$ denotes the surface normal at point $\mathbf{v}_i$ in the template mesh and $\mathbf{u}_i$ is its closest point within the target mesh. The method follows the standard ICP strategy, with the difference that the point-to-plane distance is optimized, improving robustness with respect to bad initialization. The weights w_i are intended to discard outliers and are estimated as in Sec. 3.2. At each ICP iteration, Eq. 4 is solved as in Low (2004). Note that, as in Weise et al (2011), we only used the vertices from the upper part of the face template (set U_H; see also Fig. 3c) to gain robustness against non-rigid facial deformations, e.g., when people speak.

For the overall tracking initialization, we use a face detector (Viola and Jones, 2001) to set the initial translation $\mathbf{t}_0$ (the z value is set from depth) and assume a frontal head pose (i.e., $\mathbf{R}_0$ is set to the identity $\mathbf{I}_3$). For the frame to frame case we initialize the ICP algorithm using the head pose estimation from the previous frame.

At the end of this step, for each time t, the 3D eyeball position in the **WCS** is then given by $\mathbf{o}_t^{\mathbf{wcs}} = \mathbf{R}_t \mathbf{o}^{\mathbf{h}} + \mathbf{t}_t$, where $\mathbf{o}^{\mathbf{h}}$ is the eyeball 3D center in the head coordinate system (**HCS**)[2]. Note that $\mathbf{o}^{\mathbf{h}}$ is generated automatically from the 3DMM semantics when the person specific template is created.

3.4 Eye appearance pose-rectification and alignment

Rectification. The key step for head pose invariance is the rectification of the face texture to a canonical head pose, which is done as follows. Given a textured 3D mesh (i.e., a mesh where each 3D point is associated with a RGB color) of the face image at time t, we render it after applying the rigid transformation $\mathbf{p}_t^{-1} = \{\mathbf{R}_t^\top, -\mathbf{R}_t^\top \mathbf{t}\}$, i.e., the inverse of the estimated head pose, generating a frontal-looking face image (Fig. 3d). As textured mesh, we considered two possibilities. A **data-driven mesh (*DDM*)**, obtained by mapping the RGB texture to the raw depth mesh built from the D channel of the sensor. And a **template-driven mesh (*TDM*)**, resulting from the mapping of the texture to the fitted person-specific 3DMM. Note that the rectification does not require a prior knowledge of the user's appearance and only assumes that the calibration is accurate enough to bind the RGB data to a mesh surface.

[2] To avoid repetitions, we here focus on a single eye, but the process of eye tracking and gaze estimation should be understood as done for both the left and right eye separately.

Both methods have their pros and cons (see Fig. 6 for rectification samples). We could expect a better accuracy from the *DDM*, but this is subjected to all types of sensor noise from the depth channel like the measurement noise or the absence of data due to sensing issues (e.g., when being too close to the sensor, see Experiment Section). The template approach, depending on the 3DMM fitting quality, provides a looser fit to the actual user eye 3D surface, but provides a smoother surface for the rectification and frontal rendering.

Eye Alignment. This step is illustrated in Fig. 3e-f). Thanks to the rectification, we can extract an image $\mathbf{I}^R$ around the eye region, out of which a more precise eye image could be extracted within a predefined window $\mathcal{W}$ whose position is defined by the eyeball center $\mathbf{o}^{\mathbf{h}}$.

In principle, due to the 3DMM fitting, this window should capture the same part of the eye for different users, if head pose tracking errors are not considered. However, due to the uncertainty affecting the accuracy of the 3DMM fitting, or the natural human variations in the eyeball localization, which are not perfectly correlated to the position of facial features (e.g. eye corners), this may not be the case, as illustrated in Fig. 13.

To address this issue, the parameters θ of an alignment transform are learned for each user using a small set of samples, as explained more precisely in Section 5.2. They are used to transform the window $\mathcal{W}$ into the aligned one $\mathcal{W}^A$ defining the region of $\mathbf{I}^R$ where the image $\mathbf{I}^C$ is actually cropped for further processing.

Note that the coordinate transformations in the $\mathbf{I}^R$ image domain can be directly reinterpreted within the **HCS** domain. Therefore, the alignment transform can be seen as a local transformation of the 3DMM fitted model itself, as a refinement step. Indeed, when θ defines a translation and assuming the estimated head pose is not affected by such 3DMM local refinement, the refined face model would generate the same eye image to $\mathbf{I}^C$, in particular for the *DDM* case

3.5 Gaze estimation

The pose-rectified and aligned cropped eye image $\mathbf{I}^C$ is used to estimate the gaze direction using a regression estimator. As these images are normalized, any standard method can be used, and we focus on recent appearance based methods (ABMs), which are described in more detail in Section 4.

The input to the gaze estimator is the image $\mathbf{I}^C$ and the output is the gaze direction, parametrized by the gaze yaw and pitch angles, or equivalently, by the unitary 3D vector $\mathbf{v}^{\mathbf{h}} \in \mathbb{R}^3$ defined in the head coordinate system. This vector can be transformed into the **WCS** system and used with the eye center $\mathbf{o}^{\mathbf{wcs}}$ to define the line of sight (3D ray in the **WCS**) as:

$$LoS^{\mathbf{wcs}}(l) = \mathbf{o}^{\mathbf{wcs}} + l\ \mathbf{v}^{\mathbf{wcs}}, \tag{5}$$

where $\mathbf{v}^{\mathbf{wcs}} = \mathbf{R}\mathbf{v}^{\mathbf{h}}$ and $l \in [0, \infty[$.

4 Appearance based gaze estimation methods

Thanks to the head pose rectification and alignment steps, the gaze estimation problem is simplified and we can apply any method that was originally designed for fixed head pose or head mounted cameras.

We assume that we are given a training set $\mathcal{V} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$containing N pairs of descriptors $\mathbf{x}_i \in \mathbb{R}^D$ (extracted from the cropped images $\mathbf{I}^C$), and associated gaze directions $\mathbf{y}_i \in \mathbb{R}^2$ represented by their gaze yaw and pitch angles. We also define $\mathbf{X} \in \mathbb{R}^{D\times N}$ (resp. $\mathbf{Y} \in \mathbb{R}^{2\times N}$) as the matrix where each column contains one descriptor (resp. gaze direction) from $\mathcal{V}$. The goal is to infer the gaze direction $\hat{\mathbf{y}}$ for a test sample $\hat{\mathbf{x}}$.

In the rest of this section, we focus on a baseline (kNN) and the recent state-of-the-art methods (Lu et al, 2011; Noris et al, 2010; Martinez et al, 2012) that have shown good performance and that we have implemented.

4.1 k-Nearest Neighbors (kNN)

Features. The eye image[3] $\mathbf{I}^C$ is first contrast normalized (by setting their mean to 128 and normalizing their standard deviation to 40), and all pixels are stacked into a column vector to form the descriptor $\mathbf{x}$.

Regression. The $K = 5$ nearest neighbors of the test sample $\hat{\mathbf{x}}$ (according to the euclidian distance) are extracted, and their gaze directions $\{\mathbf{y}_k\}$ are used to compute the gaze of $\hat{\mathbf{x}}$ as

$$\hat{\mathbf{y}} = \sum_{k\in\mathcal{K}} w_k \mathbf{y}_k, \tag{6}$$

where $\mathcal{K}$ contains the neighbors indices, and the weights $\{w_k\}$ are set inversely proportional to the distance to the test sample.

4.2 Adaptive Linear Regression (ALR)

This method was originally proposed by Lu et al (2011).

Features. The $\mathbf{I}^C$ image is first contrast-normalized as in the kNN case. The descriptor $\mathbf{x} \in \mathbb{R}^{15}$ is then created by dividing the image into 5×3 regions, and computing the cumulative intensity in each region (cf. Fig. 4). To

[3] Note that in all methods $\mathbf{I}^C$ is a gray-scale image of size 55×35. This is a conservative choice, since in our experiments eye image sizes almost never go beyond $\approx 20\times15$. It should however not be harmful in principle.

gain further robustness against illumination changes, the resulting values are normalized such that $\mathbf{1}^\top \mathbf{x} = 1$, where $\mathbf{1} = [1, 1, \ldots, 1]^\top$.

Regression. Estimation is formulated as a sparse reconstruction of the test sample $\hat{\mathbf{x}}$ from a linear combination (represented by $\mathbf{w}$) of the training samples $\{\mathbf{x}_i\}$. The optimal weights $\hat{\mathbf{w}}$ are obtained by solving:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \|\mathbf{w}\|_1 \quad s.t. \quad ||\mathbf{X}\mathbf{w} - \hat{\mathbf{x}}||_2 < \epsilon, \tag{7}$$

and then used to compute the test sample's gaze as $\hat{\mathbf{y}} = \mathbf{Y}\hat{\mathbf{w}}$. The implicit assumption is that enforcing sparsity will induce the selection of only a few samples within a small region of the appearance manifold, such that the same linear mapping in the appearance and gaze spaces can be exploited.

In the above formulation, the parameter ϵ plays an important role. Lu et al (2011) recommended to obtain ϵ from cross validation on the training set. However, our much noisier data drastically differ from the well controlled conditions used by Lu et al (2011). Therefore the ϵ value resulting from cross validation usually happened to be too restrictive at test time. We thus resorted to the original proposition by the same authors, where the optimal value of ϵ should be determined when the minimized $\|\mathbf{w}\|_1$ is equal to 1. In practice, we evaluated this using seven predefined values of ϵ, at the cost of longer computation time.

Finally, note as well that solving the problem in Eq. 7 is difficult, with a computation complexity increasing rapidly w.r.t. the number of training samples, thus limiting its application to small training sets. Nevertheless this was shown sufficient to obtain good-accuracy.

4.3 Multi-level HoG and Retinex Support Vector Regression (H-SVR and R-SVR)

These two methods were proposed by Martinez et al (2012) and Noris et al (2010) for head mounted camera systems, required invariance to illumination, and differ only on the feature type: Multi-level HoG (mHoG) features for the former, retinex for the latter.

mHoG Features. The image is divided into 1×2, 3×1, 3×2 and 6×4 block regions, each of which is divided into 2×2 cells from which signed HoG histograms of 9 orientations (Dalal and Triggs, 2005) are computed (see Fig. 4). The histograms are L2-normalized per block. Then $\mathbf{x}$ corresponds to all concatenated HoG histograms. Gradient features can provide robustness against illuminations issues, while histograms may lead to more robust features against noisy location of the eye region. Indeed, in the study of Schneider et al (2014)

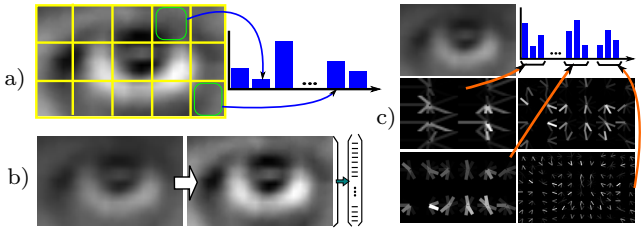

Fig. 4 Features extraction a) Descriptor used for Adaptive Linear Regression (ALR) b) Weighted retinex c) mHoG .

(on rather high resolution images), a comparison with 7 other features showed that multilevel HoG was performing best[4], with SVR (out of 6 classifiers) being the best regressor.

Retinex Features. To minimize the impact of non-uniform eye illumination variations, a retinex technique, weighted according to local contrast (Choi et al, 2007), is applied to the input image $\mathbf{I}^C$. The image pixels are then stacked in column to generate $\mathbf{x}$. Note that this feature was not tested (and thus compared with mHoG) in Schneider et al (2014).

Regression. The regression of the gaze parameters is done using a ν-Support Vector Regression (νSVR), where each gaze angle is regressed separately.

The principle of SVR is to learn a linear regression function in a high dimensional space where the input features have been implicitly mapped, and in which the scalar product between two elements i and j can equivalently be computed as $\mathbf{k}(\mathbf{x}_i, \mathbf{x}_j)$, the kernel value between the elements in their original space. The parameters are then obtained by optimizing the structural risk, allowing to find a compromise between overfitting and model complexity. As in (Martinez et al, 2012; Noris et al, 2010), we rely on the νSVR rather than ϵSVR in order to have a better control of the learning error. More details can be found in Smola and Schölkopf (2004).

The hyper-parameters of the models are $\mathbf{C}$ and ν, which controls the weights of the different costs of the objective function, and the precision γ of the Radial Basis function kernel $\mathbf{k}$ that we use. For all given experiments, these parameters were set through a 10-fold cross validation on the training data with a grid search over reasonable values.

4.4 Head pose (HP)

Finally, we include a"dummy" algorithm, which is denoted "Head pose" (HP). It corresponds to not using a gaze estimator but always setting the gaze parameters to zero, or equivalently, $\mathbf{v}^{\mathbf{h}} = [0, 0, 1]^\top$ (i.e., gazing forward). In the 3D space, this corresponds to using the

[4] except when combined with local binary patterns, although the gain in accuracy was negligible: 0.02°

head pose as gaze direction. This strategy is reported in the experimental section to convey the actual amount of gaze variations observed within our data.

5 Person invariant gaze estimation

In this section we address the person invariance problem, which we denote as the situation in which there is no training data available for the test subject in order to learn an appearance to gaze regression model.

In Section 5.1 we describe how we learn person invariant classifiers for the different gaze models of Section 4. Then, in Section 5.2, we address the cross-user eye image alignment problem.

5.1 Person invariant classifier

Joint model training. We assume that a dataset $\mathcal{V}_i$ of gaze annotated training samples processed according to the method outlined in Section 3.1 (Fig. 3) is available for each of the M subjects. The simplest strategy to acquire person invariance is to create a joint training set $\hat{\mathcal{V}} = \cup_{i=1}^{M} \mathcal{V}_i$. an approach that can immediately be applied to the kNN, R-SVR and H-SVR classifiers.

Unsupervised Adaptive Linear Regression. As mentioned in Section 4.2, an important limitation of ALR is its computation time, which prohibits the usage of $\hat{\mathcal{V}}$ as training data. To address the person invariance case, we instead propose an unsupervised selection of training sets within the database.

This is done by monitoring the reconstruction weights among the subjects found from optimizing Eq. 7. The main hypothesis is that samples from subjects which are more relevant to the current test subject are given higher weights when solving Eq. 7. In practice, a small number of test samples of the given user were used and processed with ALR using the full model $\hat{\mathcal{V}}$. The weight distribution among the set of subjects is computed. Then, the samples of the three subjects having the larger total weight were used as training data to process all samples from the test subject.

5.2 Alignment

One issue when combining data from different users is that the image cropping defined from the proposed 3D rectification may not extract exactly the same eye region. For instance, the data collected for two users may exhibit a systematic translation bias: roughly speaking, for the same gaze, in the cropped images, the iris location of the first user is systematically displaced by a few pixels form the iris location of the second user. In practice, this spatial alignment error can result in a systematic gaze angular error bias when inferring the gaze of the first user using the training data from the second user. In the next subsections, we first present a standard approach to address the alignment problem[5], and introduce our proposed alignment methodology

5.2.1 Eye corner alignment

To align eye images, the common strategy consists of locating the eye corners in a few frames, and use this information to estimate the parameters of the transformation that bring them back to a canonical position. The eye corner localization is often done manually (e.g. Martinez et al (2012)), and then the same parameters used for all frames. Automatic methods have been used but so far on high resolution images (e.g. see eye in Fig. 2d). For much lower resolution conditions such as in our data (eye in Fig. 2e) this can be problematic in terms of localization accuracy despite important recent advancements (e.g., see Kazemi and Sullivan (2014)).

Besides the localization issue, we argue this alignment strategy is not optimal for the task of gaze estimation, as discussed below. We therefore present an alternative in the next section.

5.2.2 Synchronized Delaunay Implicit Parametric Alignment (SDIPA)

Ideally, an alignment strategy aiming at person invariance should be based on aligning the eyeball positions of the different subjects, and not necessarily specific facial features such as eye corners. However, as the eyeball centers are not directly observable, we propose instead to use a direct image registration technique. In this manner the important eye structures (and in particular the iris) of different subjects gazing in the same direction are always located at the same place.

Alignment modeling. Assume we are given a set of training images $\mathcal{V}_i = \{(\mathbf{I}_k^i, \mathbf{y}_k^i), k = 1, ..., K_i\}$ for each user i. Our aim is to find for each user the parameters θ_i of a warping function $f(\mathbf{u}; \theta_i)$ registering the input images into a canonical frame. More precisely, if $\mathbf{u}$ denotes the pixel coordinates in the canonical frame, the aligned images $\tilde{\mathbf{I}}_k^i$ are then defined as

$$\tilde{\mathbf{I}}_k^i(\mathbf{u}; \theta_i) = \mathbf{I}_k^i(f(\mathbf{u}; \theta_i)). \quad (8)$$

To compute the parameters $\Theta := \{\theta_i\}_{i=1}^{M}$, we make the assumption that *when two subjects gaze in the same*

[5] Note that when the test data corresponds to the same subject than in the training set, the alignment is not needed as we may expect the cropping to be consistent between the test and training data.

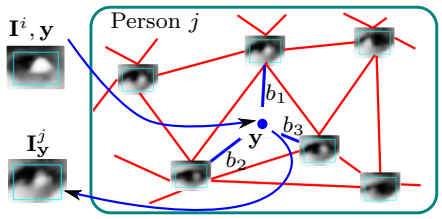

Fig. 5 Synchronized Delaunay Interpolation used to establish eye image pairs with the same gaze direction for subjects i and j.

direction, their aligned images (particularly the iris region) should match and their intensity difference should be minimal. Note that while this might not necessarily hold for all gaze directions and pairs of people, we expect this assumption to be valid on average, i.e when considering a large number of people and gaze values to constrain the parameters estimation.

However, given the image $\mathbf{I}^i_k$ of subject i with gaze $\mathbf{y}^i_k$, it is unlikely to find an image with the same gaze in $\mathcal{V}_j$. To address this problem, we propose for each $\mathbf{I}^i_k$ to use $\mathcal{V}_j$ to synthesize (as described later in this section) an image (denoted $\mathbf{I}^j_{\mathbf{y}^i_k}$) for suject j with the same gaze direction. Based on the above assumption, the alignment problem can now be defined as minimizing

$$E(\Theta) = \sum_{i=1}^{M} \sum_{k=1}^{K_i} \sum_{\substack{j=1 \\ j\neq i}}^{M} \|\tilde{\mathbf{I}}^i_k(\cdot;\theta_i) - \tilde{\mathbf{I}}^j_{\mathbf{y}^i_k}(\cdot;\theta_j)\|_2^2 + \rho R(\Theta) \quad (9)$$

where $R(\Theta) = \sum_i^M \|\theta_i - \theta^{Id}\|_2^2$ is a regularization term[6] fostering the estimation of parameters close to those of the identity warp (i.e. θ^{Id} satisfies $\mathbf{u} = f(\mathbf{u};\theta^{Id})$).

To optimize Eq. 9, we use the first order Taylor series expansion and iteratively solve for changes on the parameters from the current estimate. For efficiency, we follow a similar strategy as described in Hager and Belhumeur (1998). In this paper, we focus on the case in which the warping f represents a translation (of vector $\theta \in \mathbb{R}^2$).

Synchronized Image Synthesis. The aim is to be able to generate an eye image for any gaze parameters $\mathbf{y}$ using the training set $\mathcal{V}_j$ of subject j.

The process is illustrated in Fig. 5. In brief, we build a delaunay triangulation from the gaze angles $\{\mathbf{y}^j_k\}$ set (2-dimensional), and then find the set of vertices $\mathcal{S}^j(\mathbf{y})$ defining the triangle within which $\mathbf{y}$ falls, and generate the new image as:

$$\mathbf{I}^j_{\mathbf{y}} = \sum_{l \in \mathcal{S}^j(\mathbf{y})} b_l(\mathbf{y}) \mathbf{I}^j_l, \quad (10)$$

[6] The direct minimization of the data term is ill-posed, as the same arbitrary transform applied to all the subjects generate the same error. In practice, we used a small value of ρ to make the optimization well-posed.

where b_l denotes the barycentric cooordinates of $\mathbf{y}$ in the triangle.

Alignment Procedure. We call the method described by Eq. 9 Synchronized Delaunay Implicit Parametric Alignment (SDIPA). In the paper, we have exploited it to address two related tasks.

1. Person invariant gaze model training. In this task, the goal is to align a gaze annotated training set comprising different subjects, prior to learning the gaze regression models. We expect that exploiting aligned data will result in more accurate models. This is achieved by optimizing Eq. 9.

2. Eye image alignment for a test subject. The eye gaze model learned using the above alignment method (task 1) is person invariant, and can readily be applied to any new test subject. However, in some situations (see for instance the gaze coding experiments in Section 8), there is the possibility to gather for a test user a few samples with gaze information (e.g., a person looking at known location like another person, or simply, looking at the camera) that can be further exploited to improve the result. In this case, the same method can be used to find the eye alignment of this user with respect to the already aligned training set using these gaze annotated samples. This is simply done by adapting Eq. 9 and conducting the optimization only w.r.t. the parameters of a single subject (e.g. the θ_j of subject j considered as our test subject) while the other $\{\theta_i\}_{i\neq j}$ remain fixed. This second case can be seen as an adaptation step that is highly valuable in HRI and HHI scenarios. Notice that, even if conducting a proper gaze model training session is not possible in such scenarios, it might still be feasible to detect in a supervised or unsupervised manner instants at which the subject is fixating a given (known) target. These instances can be used to collect the few samples needed to find the test subject's alignment.

6 Experiments

In this section we first provide more details on our head pose and gaze estimation implementation. In Sections 6.2 and 6.3, we present the databases used for head pose and gaze evaluation. Finally, in Section 6.4 we describe the protocol we followed to conduct our experiments on gaze estimation.

6.1 Implementation details and speed

3D face model fitting. The 3DMM we used is the Basel Face Model (BFM) (Paysan et al, 2009). This model contains 53490 vertices and has 200 deformation

modes. The BFM was learned from high resolution 3D scans of 200 individuals (composed of 100 male and 100 female participants) thus it spans a large variety of face shapes with neutral expression.

For the face fitting step (cf. Section 3.2), we used the BFM's first 100 modes and ignored the ears and neck regions, resulting in a mesh with 41585 vertices. The γ parameter was set as $0.5\frac{N_v}{Card(L)}$, that is such that the landmarks term has 0.5 the cost of the data term, taking into account the number of data points-landmarks ratio. The λ_0 value was set empirically, such that its initial value is high enough to keep the α parameters close to $\mathbf{0}$ ($\lambda_0 = 0.1$ in our implementation) then $\lambda_n = 0.5\lambda_{n-1}$ within the iterative process.

Given a few annotated frames with landmarks (typically 1 to 5 frames), the fitting algorithm takes from 5 to 20 seconds to optimize. Note that since people face shape is not expected to change much, this step is only performed once per subject, which means that the fitted model can be reused across sessions.

Head pose tracking and rectification. Once the face model is created, we used only 1710 points from an upper face region defined a priori within the BFM topology (see purple region in Fig. 3c). The eye rectification itself is implemented as an OpenGL rendering with the viewpoint defined according to the inverse head pose.

Alignment. The warping function f used in this paper is a translation ($f(\mathbf{u};\theta) := \mathbf{u} + \theta | \theta \in \mathbb{R}^2$)) which we found sufficient to (implicitly) align the eyeball position across subjects after rectification.

Person invariant gaze model training: solving Eq. 9 to find the per-subject eye alignment parameters prior to the training of a person invariant model can take 5-10 minutes for the 16 subjects of the EYEDIAP database, using 50 images per subject. This is acceptable, as it has to be done only during the training of person invariant models from a dataset.

Eye image alignment for a test subject: for a test subject, finding her alignment parameters θ with respect to an already aligned training dataset (step above) takes around 10s from when using 1 to 5 sample images, but there is much room for improving our implementation. Importantly, note again that this has to be done only once per subject, and that the same parameters can be used for different sessions over time. Finally, once the alignment parameters have been estimated, computing $\mathbf{I}(f(\mathbf{u};\theta))$ for each frame during tracking is very fast as it only corresponds to warping a small image.

Gaze estimation. The feature extraction is implemented as described in Sec. 4. For SVR we used the scikit-learn software (Pedregosa et al, 2011). The kNN is based on a brute force search, but could clearly be improved, e.g. using a KD-Tree. The ALR method implementation used the CVXOPT software to solve Eq. 7.

Speed. Overall, the gaze tracking takes around 100ms per frame. Note however that this is a research implementation, where the most time-consuming elements are the data pre-processing (RGB-D 3D mesh creation) and the head pose tracking. The head pose tracking is CPU based and alone takes from 20 to 100ms (depending on the amount of head pose changes during consecutive frames). A careful GPU-based implementation could greatly increase the speed.

The OpenGL based rectification takes 15ms. The gaze regression computation time depends on the used algorithm. For a particular case of using 1200 training samples (e.g., for an experiment from Sec. 7.2) the kNN method takes 25ms per eye, the H-SVR method takes 15ms per eye, whereas the R-SVR method takes 11ms per eye. The speed of ALR is heavily dependent on the size of the training set, as discussed in Sec. 7.2.

6.2 Head pose experiments

The 3DMM fitting and head pose tracking are important elements that contribute at different stages of the processing pipeline (for the rectification, and for the final 3D LoS estimate) to the 3D gaze estimation. To evaluate and validate our method, we conducted experiments on two publicly available benchmarks, namely the BIWI kinect head database (Fanelli et al, 2011) and the ICT 3D head pose dataset (ICT-3DHP) (Baltrusaitis et al, 2012). Both datasets were recorded with a Microsoft Kinect at VGA resolution (RGB and Depth). The BIWI dataset was annotated using the faceshift software[7], whereas the ICT-3DHP dataset uses the Polhemus Fastrack flock of birds tracker.

In particular, to evaluate the benefit of using a face model specific to each user, we compared the head pose tracking results when using either the user specific 3DMM fitted or the mean face shape (i.e., assuming the shape parameters $\alpha = \mathbf{0}$). Results are reported in Section 7.1.

6.3 Gaze estimation dataset

For our main task, we used the publicly available EYEDIAP database[8] described in Funes Mora et al (2014) which provides data recorded using a Microsoft RGB-D Kinect (1st generation, for XBOX 360) that was further

[7] `www.faceshift.com`

[8] `https://www.idiap.ch/dataset/eyediap`

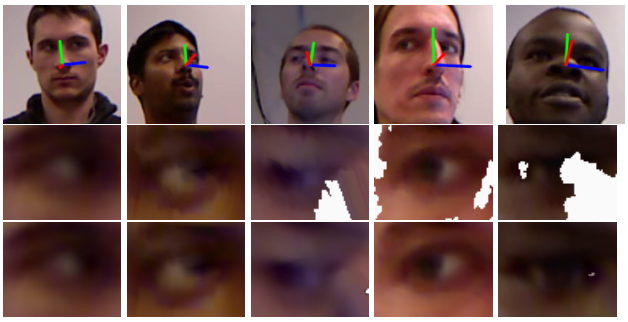

Fig. 6 EYEDIAP samples and pose-rectified images. Each column correspond to one data sample. Per row we show i) the head pose, showing a region of 140×140 pixels from the RGB frame; ii) $\mathbf{I}^R$ images generated from the depth data (*DDM*); iii) $\mathbf{I}^R$ images generated from the template data (*TDM*). The first three columns correspond to sessions involving the *FT* target, whereas the last two correspond to samples for the *CS* target.

calibrated with the method of Herrera C. et al (2012). It contains 94 recording sessions lasting between 2,5 and 3 minutes and characterized by a combination of the main variables which affect gaze estimation algorithms: different participants, ambient conditions, visual target, and head pose activity. These characteristics are summarized below.

Participants. The recordings involve 16 different people (12 men, 4 women), whose age range between 20 and 45. They are from different origin, with a total of 12 nationalities, e.g., 2 people from Central America, 1 black African, 4 caucasian French, 1 Indian, 2 caucasian Swiss, etc. As a result, the eye shapes and appearance exhibit a large variability. See samples in Fig. 6.

Visual target conditions. The recordings involved two main scenarios characterized by their gazing tasks. People had to follow either a "floating target" in the 3D space -a ball attached to a string- (*FT* condition), or they would look at a target continuously moving on a screen (*CS* condition). The *FT* case is a highly challenging problem but is interesting as it is very representative of HRI and HHI scenarios. As people were seated at a distance of $\approx 1.2m$ from the sensor, the typical depth error at this distance is around 3mm (according to Herrera C. et al (2012)), the typical eye image size is $\approx 13 \times 9$ pixels, and the gaze space is as large as $\pm 45° \times \pm 40°$. The head pose variations follow a similarly large range.

In the *CS* case, the person sat closer (at $\approx 0.9m$), leading to a typical depth calibration error of 1.5mm and a typical eye image size of $\approx 19 \times 14$ pixels. However, as the screen is well confined (spatially), the range of gaze directions (in 3D space, without considering head pose variations) is smaller, i.e. $\approx \pm 15° \times \pm 10°$. Some example images, before and after the pose-rectification procedure, are shown in Fig. 6

Head pose activity. For each of the target situations, the head pose of a person was controlled for two conditions. In the Static Pose (*SP*) case, participants were asked to keep the head approximately fixed. In the Mobile Pose (*MP*) case, they were asked to move and rotate the head in all directions (while still looking at the target), resulting in large head variations in the recorded data. Apart from that, participants were not requested to maintain a neutral expression, and the data involves people speaking or smiling.

Combined with the low eye image resolution, this makes the EYEDIAP dataset more challenging than many databases discussed in the literature while corresponding to conditions frequently encountered in HRI and HHI.

6.4 Gaze estimation experimental protocol

Ground-truth gaze. The EYEDIAP data comes with gaze information. More precisely, the 3DMM of each of the participants was fitted using the method described in Section 3.2, and for each session, the head pose was tracked using the algorithm described in Section 3.3, allowing to obtain the eyeball center $\mathbf{o}^{\mathbf{WCS}}$ of an eye. Similarly, the point of regard $\mathbf{p}_{PoR}$ was extracted in **WCS**, either by tracking the ball or by knowing the target in the 3D calibrated screen, and used to derive the ground-truth gaze unitary vector in **WCS** as $\mathbf{v}^{\mathbf{g}t} \propto \mathbf{p}_{PoR} - \mathbf{o}^{\mathbf{WCS}}$. As an indication, the location errors were estimated to be around $5mm$ on average[9] leading to an estimated accuracy of the gaze direction of around 0.25^o.

Annotations. For each session a file provides the frames that can be considered as valid. In brief, this was obtained either automatically, semi-automatically, or manually, excluding frames to account for the following: i) it is important to note that *not all* frames in a session are annotated with $\mathbf{v}^{\mathbf{g}t}$ as the EYEDIAP dataset consists of non stop video recordings. There were moments in the *FT* case where the GT could not be determined, like when the ball's position could not be retrieved as it was either out of the sensor's field of view, or so close that the sensor does not output depth data (needed to determined its 3D position). ii) for the *CS* case, each time

[9] This is an educated estimation. Location errors for the ball target or the screen dot center is considered as 0, but we needed to add the depth uncertainties or calibration errors. For the eyeball center, we evaluated the error by comparing in a few frames the manual annotation of the eyeball center with the projection of $\mathbf{o}^{\mathbf{WCS}}$.

$\mathbf{p}_{PoR}$ *randomly* starts a new trajectory (i.e., a new dot is displayed on the screen), a set of frames were systematically removed from the annotation to allow sufficient time for the gaze shift and ensure that the participant is again fixating at the target. iii) self-occlusion situation. In sessions with head pose variations, frames where the eye was not fully visible and occluded by the nose (as estimated from the head pose) were removed. iv) extreme gazes. Frames were removed in situations with head pose and $\mathbf{p}_{PoR}$ measurements, but with a $\mathbf{v}^{\mathbf{g}t}$ almost impossible anatomically (yaw beyond 45 degrees), making it unlikely that the person was actually gazing at the target. v) finally, manual inspection was conducted to eliminate frames with blinks and obvious distractions (the participant is not fixating at the visual target). Note that the criteria iii and iv were applied to each eye separately, meaning that in a given frame one eye annotation can be considered as valid while the other is not.

As a result of these validity checks, the average number of valid frames per session is around 2400.

Performance measure. At each time frame, we used the gaze angular error, computed as follows:

$$\epsilon_g = \arccos(< \mathbf{v}^{\mathbf{wcs}}, \mathbf{v}^{\mathbf{g}t} >) \quad (11)$$

where $\mathbf{v}^{\mathbf{wcs}}$ is the estimated gaze direction. Aggregated performance was obtained by computing the mean angular error over the test frames of each session. The average and standard deviations were then computed from the results obtained from the relevant sessions.

Missing measurements. When using depth measurement for the rectification (see Sec. 3.4), some pixels of the cropped image $\mathbf{I}^C$ may not be associated with any RGB measurement (see Fig. 6). This can be due to large head poses causing self occlusion, or missing depth data measurements in the eye region. To handle this situation, the gaze classification methods were updated as follows. In the kNN and ALR cases, the missing dimensions were simply excluded in the distance computation (kNN) or in the reconstruction (ALR[10]). In the R-SVR and H-SVR cases, the missing pixels were simply replaced by the average of the available measures.

Experimental protocol. In all the evaluations, the training data is disjoint from the test data. This was obtained either by training on one/several session(s), and testing on another one (e.g. for testing head pose or person invariance), or by splitting temporally a given session in two halves.

[10] As in ALR a dimension is obtained by averaging over pixels within a region, the dimension was discarded if the proportion of availble pixel values was less than 30%

Table 1 Head pose tracking mean absolute angular errors obtained for the BIWI dataset. The Regression Forest method is from (Fanelli et al, 2011) and the CLM-Z with GAVAM from (Baltrusaitis et al, 2012).

Method	Yaw	Pitch	Roll	Mean
Regression forests	9.2	8.5	8.0	8.6
CLM-Z with GAVAM	6.29	5.10	11.29	7.56
Proposed (mean shape)	4.53	2.76	3.95	3.75
Proposed (3DMM fitting)	2.43	1.91	2.67	2.34

During evaluation, valid frames were further filtered to exclude the test samples in which the gaze ground truth was not within the convex hull of the training data (in terms of gaze angles defined w.r.t. the **HCS**). This was motivated by the fact that, in the head pose invariance experiments with the screen (*CS* case, Sec. 7.3), there was a mismatch between the training and test gaze angles data at times during the sessions. Since it is known a priori that the regression methods in appearance based approaches can not handle well extrapolation to unseen gaze data[11], considering samples out of the convex-hull of the training set would introduce much (random) 'noise' in the evaluation. deflecting the reported errors from conveying the actual performance achieved by the different algorithms. In the (*CS*) case, as the screen is a small object (within the larger 3D space), the training samples collected using a static head pose only cover a small gaze space region, whereas sessions with head pose variations induced a larger coverage as the screen region would move within this space following head movements, causing the aforementioned mismatch.

For head pose invariance experiments in the *CS* case, this discarded ≈40% of the test samples. Nevertheless, the remaining samples are still diverse in terms of combined head pose and gaze directions. Note that (i) in the other experiments (*FT* target, person invariance), excluded frames represented less than 5% of the test frames and (ii), in all cases, as the training and test samples are the same across different gaze regressions methods, results accross methods are directly comparable. Protocol elements specific to a given experiment are presented in the result Section.

7 Results

In this section we describe the results obtained using our framework. Section 7.1 reports the evaluation and validation of our head pose tracking method, while Sections 7.2 to 7.5 present with more details the results

[11] In a given application, training data would need to be collected appropriately.

Table 2 Head pose tracking mean absolute angular errors obtained for the ICT-3DHP dataset

Method	Yaw	Pitch	Roll	Mean
Regression forests	7.12	9.40	7.53	8.03
CLM-Z with GAVAM	2.9	3.14	3.17	3.07
Proposed (mean shape)	4.44	2.78	4.13	3.78
Proposed (3DMM fitting)	3.61	2.25	3.61	3.16

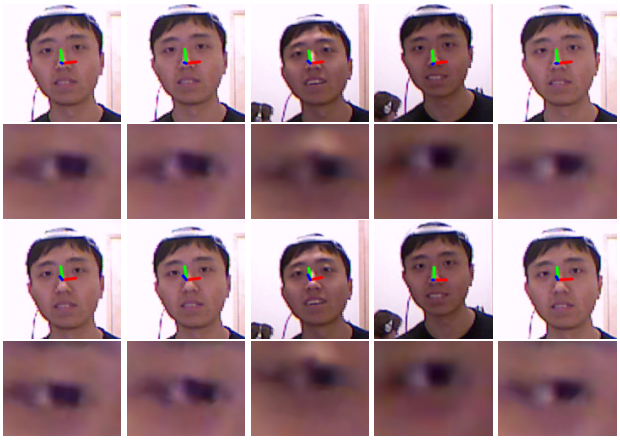

Fig. 7 Impact of using a personalized face model on the rectified eye image cropping (using the data drive rectification *DDM*). Each column depicts a different frame from a sequence from the ICT-3DHP database. The first 2 rows correspond to the results obtined when using a personalized 3DMM fitted face model, while the two last rows show the results using only the mean face shape. As can be seen, in this later case, the rectification exhibit more unconsistent eye cropping in both the vertical and horizontal directions, which would negatively impact the gaze estimation process.

of the gaze estimation experiments conducted on the EYEDIAP database, discussing the different aspects (appearance models, pose and person invariance, alignment, etc.) of our methodology.

Note that while the gaze evaluation results are summarized in Tables 3 and 4, along with the main acronyms used to report conditions, the detailed results (per single experiment) are provided in the Appendix.

7.1 Head pose tracking results

The results obtained for the head pose tracking experiments are reported in Table 1 and 2. In addition, we report the results from two alternative methods, namely Regression Forest (Fanelli et al, 2011), and CLM-Z with GAVAM (Baltrusaitis et al, 2012) which is a fitting method relying on both depth and RGB data. The performance reported for these methods was obtained from the experiments conducted by Baltrusaitis et al (2012).

We can observe our head pose tracking method has by far the lowest error for the BIWI dataset. In some sessions we encountered extreme head poses for which there were no depth measurements in the upper face region and caused the tracker to get lost deviating the error mean (recall our method tracks only this region). If we ignored 4 sessions (out of 24), the mean angular error reduces to 1.61°. However note that as the annotations for this dataset were obtained using a head pose tracking method similar to ours (faceshift, Weise et al (2011)) applied to the full face, we can only conclude our tracker obtains comparable results.

For the ICT-3DHP dataset our tracker achieves comparable results to the CLM-Z with GAVAM method. However, for a particular session, the subject's hair caused an important failure. If this session is ignored (1 out of 10) the average error further reduces to 2.68°. This suggests that a better outlier detection strategy (possibly exploiting visual data) would be beneficial in future work. Furthermore, the evaluation of our tracker got affected due to ground truth mis-synchronization. More precisely, while the ground truth has been with the RGB video, the RGB video happens to lose synchrony with depth in some sequences. This is a problem as our tracker is purely based on depth (whereas the CLM-Z GAVAM method relies as well on the RGB data), causing misleading errors, for example, during fast head movements. Nevertheless, both the BIWI and ICT-3DHP experiments demonstrate that our tracker achieves high accuracy.

Table 1 and 2 also compare the results using either the 3DMM-based personalized template or only the BFM's mean face shape. Even though using the mean shape leads to good results, using a personalized template do lead to more accurate head pose estimates.

Note that accurate pose estimation is important for our method, as pose estimation impacts gaze estimation in two ways. First, as a direct input to the estimation of the line of sight in the 3D space (see step h) in Fig. 3). In this case, an error made in the estimation almost immediately translates into an error in gaze estimation. Second, in the extraction of the cropped rectified eye images, which needs to be consistent over frames (i.e. having the image projection of the eyeball center always at the same position) for the same person, since a displacement error will translate into gaze estimation errors[12].

To qualitatively illustrate the impact on gaze tracking for this second point, we present in Fig. 7, for a representative sequence, the eye cropping resulting from using the 3DMM fitted model or the mean face shape. As can be seen, since the mean shape do not fit well the given subject, the pose tracking results oscillate even for similar head poses, generating an inconsistent frame by

[12] And the global alignment strategy only correct a systematic displacement bias error for a person, not per-frame errors.

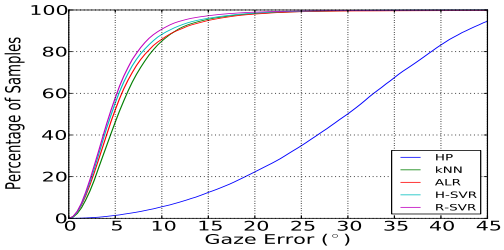

Fig. 8 Recall-error curve obtained for each of the gaze estimation methods. *FT* target, person-specific model (*PS*), minimal head pose variations (*SP*).

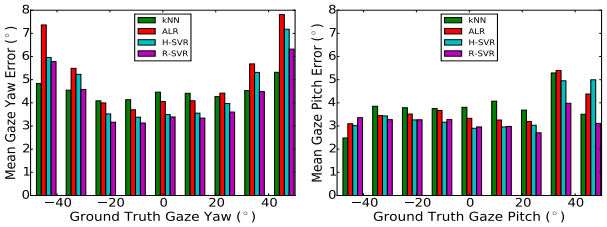

Fig. 9 Gaze error distribution in function of the ground truth gaze angles. Conditions: *FT* target, person-specific model (*PS*), minimal head pose variations (*SP*).

frame cropping of the eye image. Notice in contrast the more stable results were obtained when using a personalized face model. Thus, overall, the better tracking results validate the use of a personalized template over the simple use of the mean face shape.

7.2 Static pose and person specific conditions (*SP-PS*)

In this section we compare the regression methods assuming the model is trained and tested for the same person and under minimal head pose variations. There are 19 sessions for the *FT* target, 14 sessions for the *CS* target. In each session, the algorithm was trained using the first -temporal- half, while the evaluation was done in the second half. This means that on average, around 1200 samples are used for training[13] and around 1200 are used for testing.

Gaze accuracy. The first column in Tables 3 and 4 show the mean angular errors averaged for the relevant recording sessions using the *FT* or *CS* conditions respectively. In addition, Fig. 8 provides the recall-error curve obtained for each method for the *FT* condition.

We can first notice from the results obtained using only the head pose (HP) as gaze approximation that there are large gaze variations within the data. This variability is much larger in the *FT* case, where the target was moved in the 3D space region in front of the subjects, than in the *CS* screen gazing situation. Thus, although the gaze estimation methodology is the same in both cases, the error of the different methods is significantly lower in the *CS* (1.7 to 3°) than in the *FT* case (around 6°). This difference highlights that the choice of the task and data has a large impact on the performance, and that in general errors can not directly be compared in absolute terms without taking into account the sensor and experimental conditions. Nevertheless, as shown by Fig. 8, more than 85% of the gaze errors are below 10^o in the difficult *FT* conditions.

[13] Note that for ALR, the number of training samples was limited to 150. Otherwise the test time is prohibitively large (it is 3,5secs per sample when using 150 training samples).

When comparing the different regression methods, we can notice that the SVR methods perform better than kNN and ALR, and that, under the current experimental conditions (*SP-PS*) the retinex features show better performance than the mHoG. Note also that although there are variabilities amongst the sessions, with for instance results ranging from 1.1° to 3.2° in the *CS* case and 3.1° to 9.9° in the *FT* case using R-SVR, this R-SVR method is obtaining the best results in 15 over 19 sessions in the *FT* case, and 13 out of 14 in the *CS* case (see Appendix).

Gaze error distributions. Fig. 9 displays the estimation error distributed according to the ground truth gaze angles. HP method is not shown, but its error is equal to the absolute value of the ground truth. The plots show that errors are well distributed over the large range of gaze values. Interestingly, we can note that kNN has a flatter error distribution w.r.t. head pose. In particular, it has the lowest errors at large angles, followed by R-SVR. This plot validates an important advantage of appearance based methods in general, as these are capable of gaze estimation even when the iris is heavily occluded by the eyelid (e.g., when the person is gazing down), which is not the case of geometric based methods relying on feature tracking.

Number of training samples. We also evaluated the gaze estimation error as a function of the amount of training data. In these experiments, the training set was regularly sampled (in time) to obtain the desired number of training samples. Note that the training samples are the same for all methods, and that the test data remained the same as in previous experiments.

The results are shown in Fig. 10. Even though R-SVR is outperforming the other methods when more training data is available, ALR showed to be advantageous when using a smaller training set (less than 50 samples). However, it is disadvantageous for larger amounts of data due to the computational complexity of solving the constrained L1 minimization.

Table 3 Summary of results on mean angular gaze error (°) for the floating target conditions (*FT*). For a given experimental protocol we report, per evaluated method, the mean (top) and standard deviation (bottom) computed over all relevant sessions for the given conditions: i) *SP-PS*: static pose with person-specific gaze models (Sec. 7.2) ii) *MP-PS*: mobile pose with person-specific gaze models (Sec. 7.3) iii) *SP-PI*: static pose with person invariant model (Sec. 7.4) iv) *MP-PI*: mobile pose with person invariant model (Sec. 7.5). Acronyms: *SP* (static pose) - *MP* (mobile pose). *PS* (person specific) - *PI* (person invariant). D (*DDM* data-driven rectification) - T (*TDM* template-driven rectification). *NA* (no alignment) - *FL* (automatic eye corners detection based alignment) - *EC* (manual eye corners annotation based alignment) - *A* (SDIPA-based supervised alignment) - *A5* (SDIPA-based supervised alignment using only 5 samples for the test subjects).

	SP-PS	*MP-PS* pose invariance		*SP-PI* person invariance					*MP-PI* pose and person invariance				
Method	-	D	T	*NA*	*FL*	*EC*	*A*	*A5*	*NA*	*FL*	*EC*	*A*	*A5*
HP	28.6	23.0	23.0	28.0	28.0	28.0	28.0	28.0	23.6	23.6	23.6	23.6	23.6
	3.0	4.0	4.0	2.7	2.7	2.7	2.7	2.7	3.9	3.9	3.9	3.9	3.9
kNN	6.4	9.8	9.8	12.2	11.9	10.9	10.0	**10.0**	13.7	13.4	13.3	12.2	12.4
	1.5	2.6	2.7	3.2	3.5	2.7	2.1	1.8	3.2	4.0	3.5	2.7	2.5
ALR	6.2	11.5	10.3	13.7	-	-	-	-	-	-	-	-	-
	1.9	2.2	2.0	4.6	-	-	-	-	-	-	-	-	-
H-SVR	6.0	9.3	9.0	11.8	**11.3**	10.7	**9.8**	10.4	13.0	12.6	12.0	11.6	11.8
	1.9	2.2	2.1	3.8	3.2	2.9	3.1	3.5	2.5	3.1	2.3	2.2	2.5
R-SVR	**5.6**	**8.5**	**9.0**	**11.6**	11.6	**10.6**	10.5	11.0	**11.7**	**12.4**	**12.0**	**11.4**	**11.6**
	1.7	1.8	2.7	5.1	3.5	3.4	3.6	3.7	2.7	3.2	2.6	2.4	2.4

Table 4 Summary of results on mean angular gaze error (°) for the screen target conditions (*CS*). We report the mean (top) and standard deviations (bottom) computed over all sessions relevant for a given condition. For acronyms, see Table 3.

	SP-PS	*MP-PS* pose invariance	*SP-PI* person invariance					*MP-PI* pose and person invariance				
Method	-	T	*NA*	*FL*	*EC*	*A*	*A5*	*NA*	*FL*	*EC*	*A*	*A5*
HP	13.5	15.7	13.0	13.0	13.0	13.0	13.0	15.7	15.7	15.7	15.7	15.7
	2.9	4.2	2.5	2.5	2.5	2.5	2.5	4.2	4.2	4.2	4.2	4.2
kNN	2.9	4.2	8.7	7.8	7.6	6.6	6.6	9.8	9.0	8.6	7.6	7.8
	1.2	1.3	3.5	2.2	2.8	2.9	3.2	2.6	2.0	2.5	2.3	2.7
ALR	2.4	4.8	9.2	-	-	-	-	-	-	-	-	-
	0.9	1.7	3.9	-	-	-	-	-	-	-	-	-
H-SVR	1.9	**3.5**	**5.8**	**6.2**	**5.7**	**4.9**	**5.1**	**6.8**	**6.8**	**7.0**	**5.7**	**6.0**
	0.8	1.3	3.0	2.5	2.7	2.2	2.1	2.6	1.9	2.2	1.9	2.1
R-SVR	**1.7**	3.6	6.6	6.4	6.6	6.0	6.6	7.6	7.1	7.3	6.4	6.9
	0.8	1.4	3.1	2.7	3.6	2.6	3.5	3.3	3.0	3.2	2.4	3.3

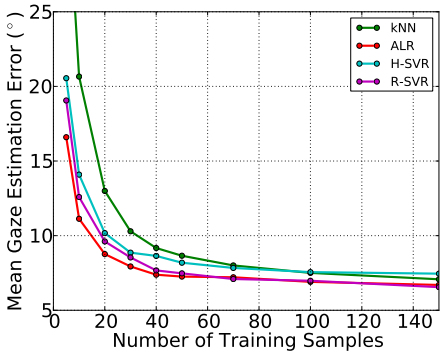

Fig. 10 Mean angular error vs. number of training samples. Conditions: *FT* target, person-specific model (*PS*), minimal head pose variations (*SP*).

7.3 Head pose invariance (*MP-PS*)

In this section we present experiments related to the head pose invariance capabilities of our framework. The specific protocol we followed here is as follows. Note that for each of the 19 (*FT*) or 14 sessions (*CS*) used in Section 7.2, there is an equivalent recording session (same person and visual target) involving head pose variations rather than a static pose. Therefore, we used as training set the session involving a static head pose and as evaluation set the equivalent session with head pose variations, each of them comprising 2400 valid samples on average. Note also that this procedure can only lead to correct results if the proposed methodology is indeed head pose invariant thanks to the generation of pose-rectified eye images.

Two rectification procedure are compared in the *FT* case: the one relying on the sensor depth data (*DDM*, D), and the one relying on the fitted template mesh (*TDM*, T), as described in Section 3.4. For the *CS* case, only the template based one could be applied. In this situation the participants were at a closer distance to the sensor, near its sensing limit, and there were too of-

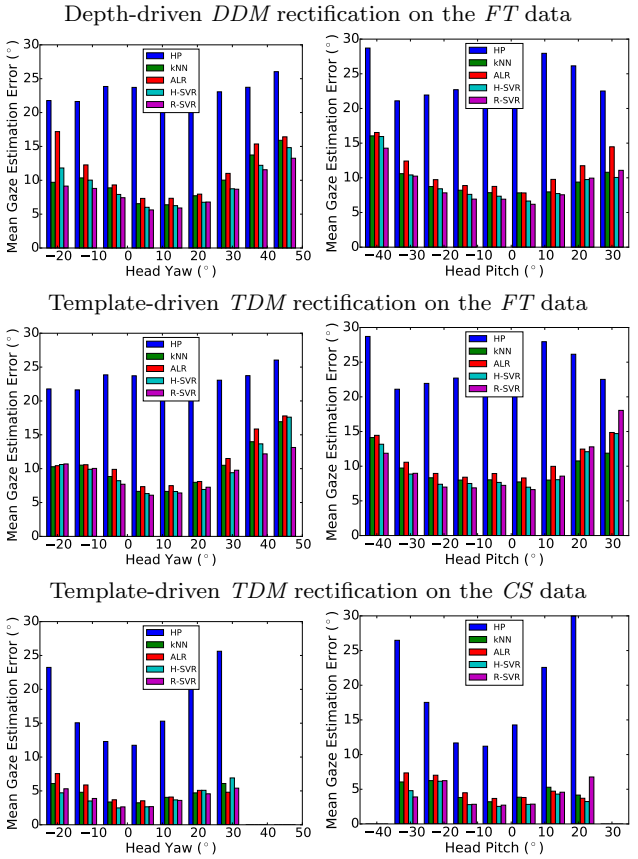

Fig. 11 Gaze error distribution for the right eye in function of the head pose for the *FT* and *CS* cases and different pose-rectification rendering methods.

ten missing depth values in the eye region (see examples in the middle row and right of Fig. 6)[14].

Results. They are shown under the "*MP- PS*" columns in Tables 3 and 4. Notice first that the HP baseline presents slightly lower error in comparison to the *SP* case (23° vs 28.6°). This is because participants tended to reduce large gaze variations when head movements were possible, hence the gaze and head pose are more correlated for these sessions. Nevertheless, the high error for HP indicates that there is still a wide range of gaze variations.

The results show a degradation of the results as compared to the static case (+3° in *FT* condition, around +1.8° in the *CS* case). This is very reasonable, considering that more than 50% of the samples have a head pose larger than 20°.

Again, the two SVR methods perform better. The ALR method seems to suffer more than the other approaches from the head pose changes. This is particularly true in the data driven case, and might be due to the loss of dimensions in the eye representation when no depth measurements are available in some eye regions. This is confirmed by comparing the error distributions according to the head pose, shown in Fig. 11: ALR errors are higher in the *DDM* case than in the *TDM* for a head yaw angle near to -10 or -20°. Notice at head yaw angles further than -20° the right eye gets more and more occluded by the nose, whereas at positive and larger angles (up to ≈ 50°) the right eye remains visible. These error distributions also show that, as expected, the errors increase in terms of the head pose angle. For our methodology, the source of errors are diverse: missing depth values, rendering artifacts due to depth-noise, and self-occlusions.

Finally note that, although the two rectification methods perform in par overall[15], the template method seems to suffer more from larger head pitch (errors near +20° and +30°) when looking up.

7.4 Static head pose, person-invariance (*SP-PI*)

To evaluate the person invariance case, we conducted a leave-one-person-out cross-validation on the sessions involving minimal head pose variations (*SP* data). This means that in each of the N experiments (where N is the number of sessions for the *FT* or *CS* case), there are around $2400 \times (N-1)$ samples available for training[16] and around 2400 for testing.

The results for *FT* and *CS* are reported under the *SP-PI* columns from Table 3 and 4 respectively, and differ on which alignment strategy was used, if any. The obtained results can be compared to the person-specific case on the same data (*SP-PS*)[17].

We evaluate four types of alignment: "*FL*" correspond to an alignment based on an automatic facial landmarks detection algorithm, "*EC*" correspond to an alignment based on manually annotated eye corners, "*A*" is our proposed synchronized delaunay implicit parametric alignment whereas "*A5*" is the same approach but using only 5 samples for the alignment of the *test* subject. *NA* correspond to no alignment.

Notice the *A* vs. *A5* comparison is motivated by possible applications where we want to reduce the load for

[14] Thus, all reported results for *CS* are with the template rectification. For the *FT*, the default rectification was with the data driven rectification *DDM*. Note that in the static pose *SP*, differences between the two rectification methods are nearly indistinguishable, as almost no 3D rotation is applied.

[15] For instance, in the R-SVR case, despite a gain of 0.5 for the depth driven approach, it only performs better than the template based method in 10 sessions out of 19

[16] For the SVR methods we limited the training set to 1200 samples as using the full set was prohibitively slow.

[17] With the slight difference that the evaluation is conducted on all samples of the test subject's session, instead of only the second half in the person-specific case.

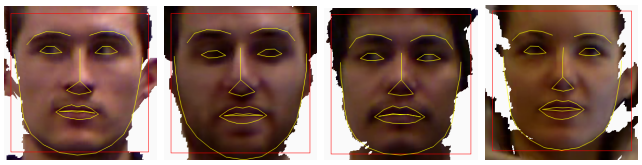

Fig. 12 Automatic landmark detection on the pose rectified face image using the method by Kazemi and Sullivan (2014).

Fig. 13 Alignment example. All samples share the same gaze direction $\mathbf{y}$. The images are shown before (top row) and after (bottom row) alignment on the dataset, for different subjects (one per column). Note for instance the discrepancy in the height location of the iris before alignment (too low in the 2nd sample, too high in the 3rd).

annotation. In this context "*A*" can be interpreted as the best case scenario whereas "*A5*" is representative of a conservative scenario where only a few gaze annotated samples can be obtained for the test subject. Please see Section 8 for an application example.

As can be noticed, there is an important error loss (around 5.5° for *FT*, 4° for *CS*). Overall, the error is larger than when considering head pose variations, suggesting that with low resolution images, eye appearance variability due to different users are more important than those due to head pose changes (even large) after our proposed rectification. The errors are not distributed equally across subjects: there are difficult cases for which the errors rise to 15.7, 17.1 and 26.4 degrees for R-SVR/*FT*, as it can be observed by its larger variance of 5.1° (*NA* case). Note that, in particular, ALR performs poorly and the mandatory selection process described in Sec. 5.1 (here obtained from 1 out of 50 samples) was prohibitively slow. For these reasons, we did not evaluate the alignment techniques with ALR.

Alignment. The first tested method is *FL*. In this case we applied the facial landmarks detection method of Kazemi and Sullivan (2014) on the pose-rectified facial images, as shown in Fig. 12; although in practice this method showed good stability on this type of images, we obtained the eye corners position for over 100 frames and computed their average to account for minor variations. Considering future improvements on automatic landmarks localization algorithms, we also evaluated the *EC* case, which means that 10 to 15 eye image samples were annotated *manually* with the eye corners[18]. In both cases this was used to register the eye images in a canonical view from the average eye corners position.

Overall, the *FL* strategy brings minor improvements to the *FT* scenario; although it has a similar behavior in the *CS* case, it actually degrades the accuracy for the H-SVR method. The gain is nevertheless larger for the *EC* strategy, with a gain of 1° in *FT*, but surprisingly almost no gain in *CS*, except for the kNN method.

Alternatively, the proposed Synchronized Delaunay Implicit Parametric Alignment (denoted *A* and *A5* in Tables 3 and 4) can be applied. In *A*, all the test gaze samples were used for alignment (including the test subject), so the method performance can be somehow considered as an oracle. In the *A5* case, 5 samples whose gaze values were close to 0 were used to align the eye of the test subject with an already aligned dataset. In practice, such samples could easily be obtained depending on the context (see Section 8 and Section 9).

The results demonstrate the interest of the method: *A* improves the result in both the *FT* and *CS* case, even outperforming the *EC* manual alignment case. Fig. 13 illustrates qualitatively the alignment effect. Despite significant differences in eye appearances, the eye alignment is visually better after *A* than before.

Finally, results of the *A5* case show that the use of a minimal set of labeled samples can bring a good gain in the result: the best performing technique in *FT* (kNN) undergoes a reduction of 2.2° as compared with no alignment (*NA*), improving the results in 18 out of 19 sessions; in *CS*, the gain is of 0.7° (for H-SVR), improving the results for 9 out of 14 sessions.

7.5 Pose variations and person invariance (*MP-PI*)

Finally, we evaluated the performance of our approach in the most general case: data with head pose variations (*MP* sessions), and using a person invariant gaze estimation model. In this case, for a test subject, the data from the static pose *SP* of all other subjects were used as training data. The size of the training and test sets are the same than in the *SP-PI* case. Similarly, the alignment parameters employed were those estimated in the static case (cf previous subsection).

The results are reported in Tables 3 and 4 under the "*MP-PI*" columns, and can be compared to those reported when using a person specific model (*MP-PS* condition). Looking at the best technique for *FT* (R-SVR) and *CS* (H-SVR), the following comments can

[18] Doing such annotation was not so easy in practice. Given our image resolution, determining visually the location of an eye corner is difficult, thus the need for multiple annotations.

be made. The person invariance situation increases the errors in a similar fashion than in the *SP* case: $+3.2°$ in *FT*, $+4°$ in *CS*. The proposed alignment approaches *A* (resp. *A5*) contribute to reduce the error: $-0.3°$ (*A5*, $-0.1°$) in *FT* for the best performing method (note that the decrease is larger with the other methods, around $-1.2°$ for H-SVR or kNN in the *A5* case); $-1.1°$ (*A5*, $-0.7°$) in the *CS* case.

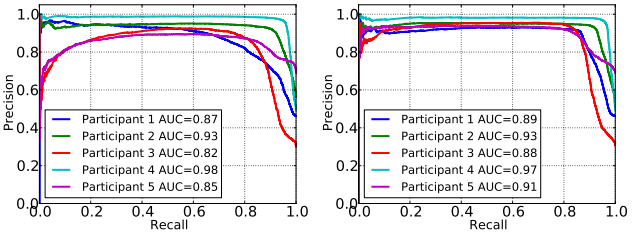

Fig. 15 Automatic gaze coding precision-recall curves obtained without alignment (left) and using alignment (right).

8 Gaze coding experiments in job interviews

In this section we describe experiments on the problem of automatic gaze coding in natural dyadic interactions.

Setup and dataset. For our evaluations, we used the SONVB dataset (Nguyen et al, 2014) which consists of real job interviews. Each person (interviewer and interviewee) is recorded using a Microsoft Kinect, as shown in Fig. 14 (top). Annotations are available at the frame level for 5 interviews, for a section of 5 minutes each.

In (Funes Mora et al, 2013), we have proposed a method to compute the relative camera pose between the two Kinects such that the **WCS** is well defined. Thanks to our method that allows to compute the Line-of-Sight (LoS) in the 3D space, we can apply a simple geometric formulation to determine whether a participant gazes at the other: provided the estimated LoS, i.e. $\{\mathbf{v}^{\mathbf{WCS}}, \mathbf{o}^{\mathbf{WCS}}\}$, and the position $\mathbf{p}$ of the visual target (i.e. the other person's head pose), we detect a gazing event when the angle between $\mathbf{v}^{\mathbf{WCS}}$ and the *reference* vector $\mathbf{v}_c := \mathbf{p} - \mathbf{o}^{\mathbf{WCS}}$ is below a threshold τ.

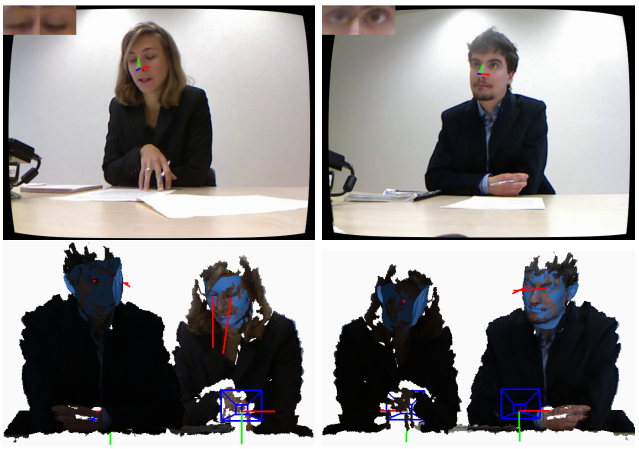

Fig. 14 Gaze coding in a dyadic interaction. Top: original RGB frames. Bottom: 3D rendering of the composed 3D scene from two viewpoints, including the estimated head pose and 3D gaze direction.

Note this scenario does not allow for a gaze calibration phase and, due to the participant's natural behavior, the head movements are unconstrained.

Protocol and evaluation. In these experiments, our main aim is to evaluate the accuracy provided by our model, including the effect of the eye alignment step. Given the results obtained in the previous section, we decided to use the H-SVR model trained from all the participants of the EYEDIAP database, as it performed the best under most conditions.

Alignment experiments. Note that the EYEDIAP training set is already aligned prior to the training phase using our method. In addition, note that at any given time, the reference vector $\mathbf{v}_c$ is known. Therefore, if it is known that one participant gazes at the other one, a gaze annotated sample for alignment can be defined from $\mathbf{v}_c$. This is very simple to do from an operational point of view: press a button at a moment when the subject gazes at the other. Here we collected only 3-5 samples per person and aligned the test subject's eye using our proposed method described in Sec. 5.2.2.

The head pose and 3D gaze direction is estimated using our proposed methodology (cf. Section 3 and Fig. 3; visual results shown in Fig. 14, bottom), from which we compute the classification score as the angle between $\mathbf{v}_c$ and $\mathbf{v}^{\mathbf{WCS}}$.

Results. In Fig. 15 we show the precision-recall curves obtained by varying τ, from which we can observe the improvement given by our alignment approach. From these curves we obtained the F_1 scores at the Equal-Error Rate point, shown in Table. 5.

This table confirms the advantage of our proposed framework. In average, the classification accuracy increases when alignment is used and, while the increase applies to all people, there are particular cases (e.g. Participant 1) where an important alignment correction was indeed necessary and the method was able to find it from very few annotated samples.

Finally, Fig. 16 shows the resulting coding at frame level as a time sequence. Notice, despite the subtle gaze behavior, the estimates follow closely the ground truth.

Table 5 F_1-score for frame level gazing detection.

	Participant					
Method	1	2	3	4	5	Mean
Not aligned	80.9	91.2	83.9	95.7	85.3	87.4
Aligned	87.4	92.3	86.7	96.0	86.9	89.9

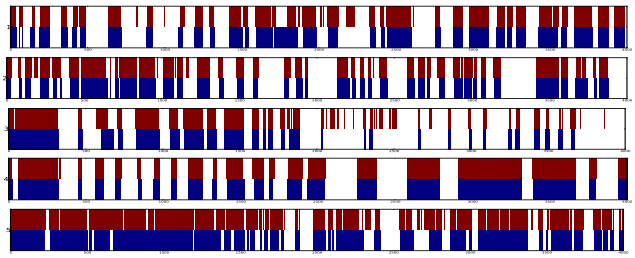

Fig. 16 Automatic gaze coding results for a sequence of ≈2 minutes. Blue: ground truth. Red: Estimated gaze coding. From top to bottom: participants 1 to 5.

9 Discussion and future work

We have proposed methods to address the head pose and person invariance problems of automatic gaze estimation and have validated them through extensive experiments. In this section, we present and discuss the limitations of the the proposed methodology and how it could be extended and improved in diverse ways in future work.

The use of a personalized (3DMM fitted) face model was shown beneficial for accurate head pose tracking and consistent eye image cropping. The required manual landmarks annotation at training is in practice a simple procedure which has to be done only once per subject. Nevertheless, this step could leverage state of the art landmarks detection algorithms to make it fully automatic, such as Dantone et al (2012), Cao et al (2013) or, as used in Sec.7.4, Kazemi and Sullivan (2014). Although such detectors may also introduce noise, keeping the more consistent landmarks results could allow the fitting to be obtained automatically and online.

Furthermore, during head pose tracking, the same facial landmarks estimates could further constrain the ICP cost function to improve the tracking accuracy, esp. in near frontal poses. Fusion algorithms and experiments would then be needed to evaluate whether the landmark extraction is robust and precise enough and can lead to the reduction of the gaze tracking errors.

An exhaustive comparison in terms of features and regression algorithms has not been conducted in this paper, as our purpose was but to validate our contributions using the best and representative features and algorithms found in the literature, as motivated in Section 5. This leaves room for future studies evaluating whether in our framework and scenarios, other types of features such as local binary patterns, the possible exploitation of color information, or combination of features (as done by Schneider et al (2014) for instance), could improve the results. In this direction, it could also be relevant to evaluate whether there is an impact of the pose rectified eye image size on the performance error, taking into account the distance at which the system is expected to operate; or similarly, evaluate the impact on accuracy of the amount of training data, e.g. by using less than 15 people, or by collecting more data to see at which level the method saturates. Such studies could be facilitated and compared to our work thanks to the use of our using publicly available database.

The alignment, which was implictly defined within a person's head frame, was intended to correct eye image cropping inconsistencies across subjects, when building a person invariant gaze model. A benefit of the method was that it can compensate for 3DMM fitting semantic errors across subjects, even if the the eye corners are not well located or difficult to locate given the image resolution. Importantly, by exploiting the gaze input to conduct the alignment, the methods implicitly strive to align the actual eyeball positions across subjects, which is what the gaze alignment step should aim for[19].

In this paper, a single alignment was performed per subject. However, in practice, we do observe as well frame to frame misalignment errors coming from small ICP fitting differences across frames, due to missing or noisier depth information (e.g., at larger depth), face deformation in the eye region, or erroneous pose estimation. To handle this, it would be interesting in future work to explore frame by frame alignment methods, e.g., through eye image stabilization leveraging on robust optical flow estimation, image registration, and landmark detection methods.

Finally, note that even though the alignment function f we used here is a translation, we hypothesize that other transforms may consider more geometric variability. In particular, including a scale may model eyeball size variations.

The eye image pose rectification plays a role as well in our approach, esp. when the eye goes towards more profile views. The *TDM* template method would profit from a 3DMM model with a tighter fit in the eye region. Local non-rigid registration methods, or unsupervised frame matching and averaging could be used there. As the depth noise level makes this challenging, RGB information could be exploited as well. Also, depth information could help handling self occlusion by the nose. The *DDM* depth driven method could alternatively make use of depth filling methods and depth smoothing, to

[19] and is different than aligning the eye corner features

maximize the region with texture information in the pose rectified image, and to reduce artifacts (see Fig. 6). Note the *TDM* approach implicitly has this function.

Finally, we want to emphasize that our proposed approach could be exploited in diverse manners for many applications. It could be used with no cooperation from the user whatsoever, meaning the overall system and person invariant gaze models are used as is, for a new test subject. Alternatively, a minimal cooperation protocol could be defined to obtain the needed alignment data, either explicitly, e.g. requesting the participant to fixate at the camera for a few seconds (Oertel et al, 2014), or implicitly through an agent (e.g. a robot) persuading the subject to do such actions either by a direct request or by leveraging on gaze priors on non verbal human behaviors in a dialogue situation. In another direction, a third person could annotate higher level gaze semantics (people gazing at known targets) as was shown in the previous Section.

10 Conclusion

In this paper we have proposed a framework for the automatic estimation of gaze in a 3D environment. We address two of the main factors which directly influence the eye image appearance, and which lead to a decrease of the gaze estimation accuracy: i) head pose variations and; ii) inter-user appearance variations.

For the challenge of head pose variations, we have proposed a framework which rectifies the captured eye images into a canonical viewpoint. To this end, we rely on depth information to accurately track the 3D head pose. Given an accurate head pose, we proposed, and evaluated two strategies for the viewpoint correction: either based on the depth measurements or the fitted 3D facial mesh.

To address person invariance, we have conducted extensive experiments evaluating state-of-the-art appearance based gaze estimation algorithms within our framework under several conditions. We have also addressed the problem of eye image alignment as it has a direct link with the person invariance problem. We therefore proposed a new method for the inter-subject eye image alignment from gaze synchronized samples, and we validated its advantage with respect to other strategies.

Finally, we have demonstrated the important potential of our system by addressing the problem of automatic gaze coding in natural dyadic interactions. We believe the proposed solution is highly valuable in many other types of scenarios in human human interaction or in human robot interaction applications.

References

Amberg B, Romdhani S, Vetter T (2007) Optimal Step Nonrigid ICP Algorithms for Surface Registration. In: IEEE Computer Vision and Pattern Recognition, pp 1–8, DOI 10.1109/CVPR.2007.383165

Amberg B, Knothe R, Vetter T (2008) Expression invariant 3D face recognition with a Morphable Model. In: Int. Conf. on Automatic Face and Gesture Recognition, IEEE, pp 1–6

Baltrusaitis T, Robinson P, Morency LP (2012) 3D Constrained Local Model for Rigid and Non-Rigid Facial Tracking. In: Computer Vision and Pattern Recognition (CVPR), Providence, RI

Baluja S, Pomerleau D (1994) Non-Intrusive Gaze Tracking Using Artificial Neural Networks. Tech. rep., CMU

Barron J, Malik J (2013) Shape, Illumination, and Reflectance from Shading. Tech. Rep. UCB/EECS-2013-117, University of California, Berkeley

Cao X, Wei Y, Wen F, Sun J (2013) Face Alignment by Explicit Shape Regression. International Journal of Computer Vision

Choi DH, Jang IH, Kim MH, Kim NC (2007) Color Image Enhancement Based on Single-Scale Retinex With a JND-Based Nonlinear Filter. In: ISCAS, IEEE, pp 3948–3951

Dalal N, Triggs B (2005) Histograms of Oriented Gradients for Human Detection. In: Computer Vision and Pattern Recognition (CVPR), vol 2, pp 886–893

Dantone M, Gall J, Fanelli G, Gool LV (2012) Real-time Facial Feature Detection using Conditional Regression Forests. In: CVPR

Egger B, Schonborn S, Forster A, Vetter T (2014) Pose Normalization for Eye Gaze Estimation and Facial Attribute Description from Still Images. In: German Conference on Pattern Recognition

Fanelli G, Weise T, Gall J, Gool LV (2011) Real Time Head Pose Estimation from Consumer Depth Cameras. In: Symposium of the German Association for Pattern Recognition (DAGM)

Funes Mora KA, Odobez JM (2012) Gaze Estimation From Multimodal Kinect Data. In: Computer Vision and Pattern Recognition, Workshop on Gesture Recognition, pp 25–30

Funes Mora KA, Odobez JM (2013) Person Independent 3D Gaze Estimation From Remote RGB-D Cameras. In: IEEE Int. Conf. on Image Processing

Funes Mora KA, Nguyen LS, Gatica-Perez D, Odobez JM (2013) A Semi-Automated System for Accurate Gaze Coding in Natural Dyadic Interactions. In: Int. Conf. on Multimodal Interaction

Funes Mora KA, Monay F, Odobez JM (2014) EYEDIAP: A Database for the Development and Evaluation of Gaze Estimation Algorithms from RGB and RGB-D Cameras. In: Symposium on Eye tracking Research & Applications

Guestrin ED, Eizenman M (2006) General theory of remote gaze estimation using the pupil center and corneal reflections. Transactions on bio-medical engineering 53(6):1124–33

Hager GD, Belhumeur PN (1998) Efficient region tracking with parametric models of geometry and illumination. IEEE Trans on Pattern Analysis and Machine Intelligence 20(10):1025–1039

Hansen DW, Ji Q (2010) In the eye of the beholder: a survey of models for eyes and gaze. IEEE Trans on Pattern Analysis and Machine Intelligence 32(3):478–500

Herrera C D, Kannala J, Heikkilä J (2012) Joint Depth and Color Camera Calibration with Distortion Correction. IEEE Trans on Pattern Analysis and Machine Intelligence 34(10):2058–2064

Ishikawa T, Baker S, Matthews I, Kanade T (2004) Passive Driver Gaze Tracking with Active Appearance Models. In: Proc. World Congress on Intelligent Transportation Systems, pp 1–12

Jianfeng L, Shigang L (2014) Eye-Model-Based Gaze Estimation by RGB-D Camera. In: Computer Vision and Pattern Recognition Workshops, pp 606–610

Kazemi V, Sullivan J (2014) One Millisecond Face Alignment with an Ensemble of Regression Trees. In: Computer Vision and Pattern Recognition (CVPR)

Li D, Winfield D, Parkhurst DJ (2005) Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches. In: Computer Vision and Pattern Recognition Workshops, IEEE, vol 3, p 79

Low K (2004) Linear Least-Squares Optimization for Point-to-Plane ICP Surface Registration. Tech. Rep. February, University of North Carolina at Chapel Hill

Lu F, Sugano Y, Takahiro O, Sato Y, Okabe T (2011) Inferring Human Gaze from Appearance via Adaptive Linear Regression. In: International Conference on Computer Vision (ICCV)

Lu F, Sugano Y, Okabe T, Sato Y (2012) Head pose-free appearance-based gaze sensing via eye image synthesis. In: IEEE International Conference on Pattern Recognition

Lu F, Okabe T, Sugano Y, Sato Y (2014a) Learning gaze biases with head motion for head pose-free gaze estimation. Image and Vision Computing 32(3):169–179

Lu F, Sugano Y, Okabe T, Sato Y (2014b) Adaptive Linear Regression for Appearance-Based Gaze Estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence 36(10):2033–2046

Martinez F, Carbone A, Pissaloux E (2012) Gaze estimation using local features and non-linear regression. In: Int. Conf. on Image Processing, pp 1961–1964

Moriyama T, Cohn J (2004) Meticulously detailed eye model and its application to analysis of facial image. Int Conf on Systems, Man and Cybernetics 1:629–634

Murphy-Chutorian E, Trivedi M (2008) Head Pose Estimation in Computer Vision: A Survey. In: IEEE Trans. on Pattern Analysis and Machine Intelligence

Nguyen LS, Frauendorfer D, Schmid Mast M, Gatica-Perez D (2014) Hire Me: Computational inference of hirability in employment interviews based on nonverbal behavior. IEEE Transactions on Multimedia

Noris B, Benmachiche K, Billard A (2008) Calibration-Free Eye Gaze Direction Detection with Gaussian Processes. In: International Conference on Computer Vision Theory and Applications, pp 611–616

Noris B, Keller J, Billard A (2010) A wearable gaze tracking system for children in unconstrained environments. Computer Vision and Image Understanding pp 1–27

Oertel C, Funes Mora KA, Sheikhi S, Odobez JM, Gustafson J (2014) Who Will Get the Grant? A Multimodal Corpus for the Analysis of Conversational Behaviours in Group. In: International Conference on Multimodal Interaction, Understanding and Modeling Multiparty, Multimodal Interactions Workshop

Paysan P, Knothe R, Amberg B, Romdhani S, Vetter T (2009) A 3D Face Model for Pose and Illumination Invariant Face Recognition. In: Proceedings of Advanced Video and Signal based Surveillance, IEEE

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12:2825–2830

Schneider T, Schauerte B, Stiefelhagen R (2014) Manifold Alignment for Person Independent Appearance-based Gaze Estimation. In: International Conference on Pattern Recognition (ICPR), IEEE

Shotton J, Fitzgibbon A, Cook M, Sharp T, Finocchio M, Moore R, Kipman A, Blake A (2011) Real-time Human Pose Recognition in Parts from Single Depth Images. In: Computer Vision and Pattern Recognition (CVPR), pp 1297–1304

Smola AJ, Schölkopf B (2004) A Tutorial on Support Vector Regression. Statistics and Computing 14(3):199–222

Sugano Y, Matsushita Y, Sato Y, Koike H (2008) An incremental learning method for unconstrained gaze

estimation. In: ECCV, Springer, pp 656–667
Sugano Y, Matsushita Y, Sato Y (2014) Learning-by-Synthesis for Appearance-based 3D Gaze Estimation. In: Computer Vision and Pattern Recognition (CVPR), IEEE
Tan KH, Kriegman DJ, Ahuja N (2002) Appearance-based Eye Gaze Estimation. In: IEEE Workshop on Applications of Computer Vision, pp 191—-
Timm F, Barth E (2011) Accurate Eye Centre Localisation by Means of Gradients. In: Int. Conf. on Computer Vision Theory and Applications, pp 125–130
Valenti R, Gevers T (2012) Accurate eye center location through invariant isocentric patterns. IEEE Trans on Pattern Analysis and Machine Intelligence 34(9):1785–1798
Viola P, Jones M (2001) Robust Real-time Object Detection. In: International Journal of Computer Vision
Weise T, Bouaziz S, Li H, Pauly M (2011) Realtime performance-based facial animation. ACM Transactions on Graphics (SIGGRAPH 2011) 30(4):1
Williams O, Blake A, Cipolla R (2006) Sparse and semi-supervised visual mapping with the S3GP. In: Computer Vision and Pattern Recognition, pp 230–237
Xiong X, Liu Z, Cai Q, Zhang Z (2014) Eye Gaze Tracking Using an RGBD Camera: A Comparison with a RGB Solution. In: International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, pp 1113–1121
Yamazoe H, Utsumi A, Yonezawa T, Abe S (2008) Remote gaze estimation with a single camera based on facial-feature tracking without special calibration actions. In: Symposium on Eye tracking Research & Applications, ACM, vol 1, pp 245–250
Yuille AL, Hallinan PW, Cohen DS (1992) Feature extraction from faces using deformable templates. International Journal of Computer Vision 8(2):99–111