A

Assessment of accuracy for target detection in 3D-space using eye tracking and computer vision

MARTIN LEROUX, Polytechnique Montréal, Rehabilitation Chair of Engineering Applied to Pediatrics
SOFIANE ACHICHE, Polytechnique Montréal
MAXIME RAISON, Polytechnique Montréal, Rehabilitation Chair of Engineering Applied to Pediatrics

Over the last decade, eye tracking systems have been developed and used in many fields, mostly to identify targets on a screen, i.e. a plane. For novel applications such as the control of robotic devices by the user vision, there is a great interest in developing methods base on eye tracking to identify target points in free three dimensional environments. The objective of this paper is to characterise the accuracy the eye tracking and computer vision combination that was designed recently to overcome many limitations of eye tracking in 3D space. We propose a characterization protocol to assess the behavior of the accuracy of the system over the workspace of a robotic manipulator assistant. Applying this protocol to 33 subjects, we estimated the behavior of the error of the system relatively to the target position on a cylindrical workspace and to the acquisition time. Over our workspace, targets are located on average at 0.84 m and our method shows an accuracy 12.65 times better than the calculation of the 3D point of gaze. With the current accuracy, many potential applications become possible, such as visually controlled robotic assistants in the field of rehabilitation and adaptation engineering.

Additional Key Words and Phrases: Eyetracking, Computer vision, 3D environment, Control paradigm, Rehabilitation

1. INTRODUCTION

Over the last decade, eye tracking systems have been developed and used in many fields such as in driving simulators [Alberti et al. 2012], in human computer interaction [Majaranta and Bulling 2014] in medical applications to improve teaching [Chetwood et al. 2012] and teamwork [Bogdanova et al. 2015], in marketing [Rosbergen et al. 1997], standard oculography [van der Geest and Frens 2002; Weber et al. 2009] and many more [Popelka et al. 2012; Staudte and Crocker 2008; Duchowski 2002]. These applications attest the large variety of information that can be interpreted from eye tracking data. However, most of those applications require a physical support, such as a screen, at a known position to work properly [Alberti et al. 2012; Popelka et al. 2012; Duchowski 2002]. This is because, given the gaze direction and the gaze plane location, it becomes trivial to compute the gaze point.

Other fields would benefit from the intuitive and powerful source of data provided by eye tracking, notably in the fields of robotic control [Jan and Bukhari 2009; Atienza and Zelinsky 2003; Lupu and Ungureanu 2013]and assistive technologies [Frisoli et al. 2012; Atienza and Zelinsky 2005]. However, these cannot generally be implemented using a fixed gazing target plane given that their target is often in an uncontrolled environment. For this reason, recent research focused on the development of methods and strategies to use eye tracking in free three dimensional (3D) environments [Fujiyoshi et al. 2012; Essig et al. 2006; Duchowski et al. 2002]. To implement applications in 3D, research works rely generally on one of the two following strategies.

(1) The 3D point of gaze approach (3DPOG) relies on the calculation of the point where the gaze vector of both eyes cross each other [Jan and Bukhari 2009; Lupu and Ungureanu 2013; Fujiyoshi et al. 2012; Essig et al. 2006; Duchowski et al. 2002]. Almost every research group using this method proposes some degree of refinement to the technique to improve accuracy. However, it is worth noting that very few papers [Abbott and Faisal 2012; Hennessey and Lawrence 2009]. describe the workspace considered during the accuracy assessment of their method, thus leaving little possibility for comparison between methods. Unfortunately, as the dis-

tance between the user and the target increases, left and right gaze directions tend to become parallel, leading to important errors in the point of gaze estimation, which limits the usable workspace to a small one that has to be very close to the user.

(2) In the mapping approach, a complementary device, such as the Kinect (Microsoft, USA) or another stereovision system with a SLAM algorithm [Paletta et al. 2013] is used to map the environment. The eye tracking data is only used to select the target inside the field of view of the said device [Frisoli et al. 2012; Atienza and Zelinsky 2003; 2005; Shimizu and Fujiyoshi 2011]. Thanks to the accuracy of stereovision systems [Peng 2011; Andersen et al. 2012; Microsoft 2016], this method is more suitable for large workspaces. In this method however, eye tracking data is underused since it is not involved in the computation of the target location which is calculated by advanced and bulky stereovision algorithms. Stereovision systems can be costly in computing power and this somewhat limits the portability or speed of the system as a whole. They are also complex and cumbersome to implement in any application without relatively advanced knowledge of image acquisition systems and algorithms. Moreover, the accuracy of stereovision systems is also very sensible to calibration.

Other similar work exists when discussing eye tracking and interaction with the field of view of the subject, such as remote eye contact detection [Ye et al. 2012] or gaze following cameras [Schneider et al. 2005]. The authors of [Pfeiffer et al. 2008] follow the gaze of a subject in a 3D virtual environment using machine learning. However, this paper focuses on estimating the location of targets in a real 3D environment, so such work cannot provide comparison benchmark in the matter.

Therefore today, most methods enabling to estimate targeted 3D locations using eye tracking have some degree of inconvenience relative to the volume or location of the functional workspace or the portability and complexity of the whole system. However, in [Leroux et al. 2015], eye tracking data was combined with a single camera to circumvent most of those limitations, but only rudimentary accuracy assessment was done. A more in depth study of the behaviour of the error and the size of the effective workspace would be essential for practical applications.

Consequently, the objective of this paper is to characterise the behaviour of the error of the method described in [Leroux et al. 2015] in order to thoroughly assert its accuracy, determine its useful workspace and identify the causes of the errors in order to improve the method in future research.

In the following sections, we first briefly recall how the eye tracking and computer vision combination (ETCV) method works and then explain how we characterized the behaviour of the error over a large workspace. Note that our workspace was chosen arbitrarily to correspond roughly to the reach of a basic 6 degree-of-freedom (DOF) serial robotic manipulator, the JACO robotic arm (Kinova, Canada) as a first application example.

2. METHOD

2.1. The combination of eye tracking and computer vision (ETCV) method

Prior any target location measurements can be made, some characteristics of the target and the system must be known. The color of the target will be used for color recognition, while the target dimensions (d) and the characteristic number of pixels per degree (PPD) of the camera will be used to compute the distance to the target assuming a pinhole camera model (see Fig. 1). These informations are stored in the memory of the system for the calculations that ensue.

While the user stares at the target, eye tracking data is acquired using the eye tracker and the associated software. We used faceLAB 5 (Seeing Machines, Australia),a remote eye tracker with a rotational tracking accuracy of $\pm 1^{\circ}$, but other commercially available devices would also be compatible with the method. Meanwhile, a snapshot is taken with a webcam fixed over the subject's head (here on a hat), sharing his field of view.

First, the eye tracking data is filtered by removing every sample where both eyes of the subject are not completely visible or undergoing a saccade - a quick involuntary eye movement. With our setup, the blinking and saccade detection indicators as well as a general quality indicator are all automatically generated by faceLAB 5 during the acquisition. Then the gaze direction, pitch and yaw angles as given by the eye tracking software, for each eye is averaged over all samples to reduce the error associated with the impossibility for any subject to stare exactly at a single small point in space knowing that the eyes are proprioceptive sensors that detect many points around the vision target. The gaze direction of both eyes is then approximated as parallel, based on the hypothesis that the target is normally far enough from the user, hence a single direction assumed to originate from a point situated exactly between both eyes is computed, thus giving the angular coordinates of the target in a spherical coordinate system originating at this point.

Secondly, previously acquired knowledge of the target color is used to locate the target on the snapshot taken by the webcam with a color recognition algorithm, a method chosen for its very simple implementation. The dimensions in pixels (d_{px}[pixels]), of the target on the picture are then measured. As can be seen in Fig. 1, the distance (R [m]) between the camera and the target can be calculated using the pinhole camera model and Equation 1.

$$R = \frac{d}{2\tan\frac{d_{px}}{2PPD}} \tag{1}$$

where R is the depth of the target, d is the target size, d'_{px} is the target image size and PPD is the Pixels Per Degree characteristic of the camera.

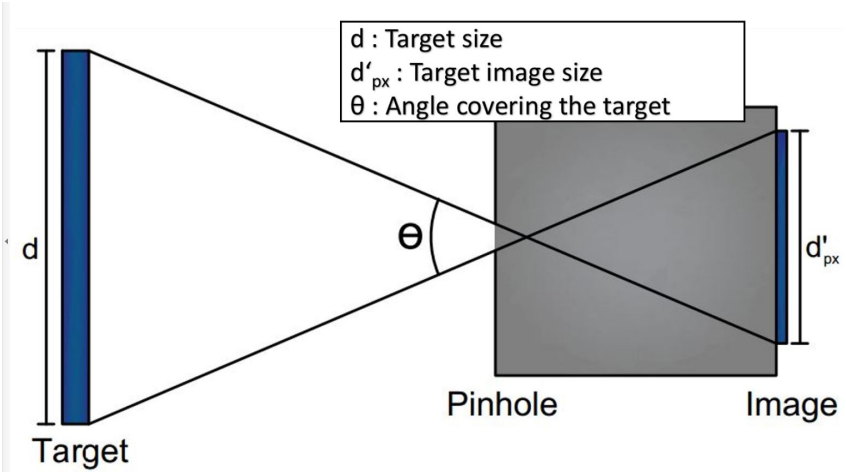

Fig. 1. Geometric representation of the pinhole camera model

We approximate R to be the same as the distance between the target and the previously mentioned origin of the spherical coordinate system between the eyes, thus completing the location of the target relatively to the subjects head. The location of the target in the eye trackers coordinate system can then be computed by taking the mean of the location of each eye, which is also given by the eyetracking data.

2.2. Validation protocol

The eye tracker was fixed on a table, on the edge of the workspace under test closest to the subject. The subjects were positioned to have their head in the eye tracker stereovision system convergence point (see Fig. 2). For most of the subjects, this meant that the eye tracker was located right above their knees. Adjusting the subjects' position rather than the eye tracker makes easier the calibration before each trial. The eye tracking system was recalibrated using the manufacturer's procedure. Computer vision data was acquired with a C016 HD 5 Mpx webcam (TeckNet, UK) fixed on a hat on the subject's head, so that it would be positioned right next to the subect's eyes and oriented so that it would share the field of view of the subject without obstructing it.

Four different variables were tested in terms of their impact on the global error, namely angular, radial and vertical position of the target, as well as the acquisition time. Each variable was tested with specific series of target positions over a cylindrical coordinate workspace. The origin of the cylindrical workspace coincides with the origin of the eye tracker coordinate system, 5.0 cm above the table. To avoid considering eventual fatigue effects, the subjects had a 2-min rest period between each serie. One variable was changed at a time, resulting in four series of 5 to 13 acquisitions(see Fig. 3):

Angular coordinate series of locations.
Thirteen target locations, from 0°to 180°, with 0°defined to right of the subject, with 15°increment, at a radial position of 40 cm and a vertical position of 0.5cm above the table.
Radial coordinate series.
Seven locations, from 10 cm to 70 cm, with 10 cm increment, at an angular position of 90°and a vertical position of 0.5 cm.
Vertical coordinate series.
Six locations, from 10 to 60 cm, with 10 cm increment, at an angular position of 90°and a radial position of 30 cm.
Acquisition time series.
Five trials, from 2 s to 10 s, with a 2 s increment, all taken at an angular position of 90°, radial position of 30 cm and vertical position of 0.5 cm.

A 2.2 cm x 2.2 cm target was placed on an interest point , facing the subject who was asked to stare at its center. A snapshot is then taken using the webcam. Finally, eye tracking data was acquired at 60 Hz for 5 seconds, except when acquisition time was under test. This amount of time was deemed sufficient based on preliminary tests.

As cited above, the workspace under test for the experience was chosen to correspond roughly to the reach of a JACO robotic arm (Kinova, Canada) used in our research laboratory for future applications, which is larger than the workspace under study in other 3D eye tracking related research that will be used as performance benchmarks.

The characterization setup is illustrated in Fig. 2.

The target registration error (TRE), defined as the Euclidian distance between the estimated target location and its actual location on the grid, was calculated for each trial. For comparison purposes (3DPOG vs. ETCV), for every trial, using the same eye tracking samples, we also reported the TRE of the 3DPOG estimated by faceLAB 5.

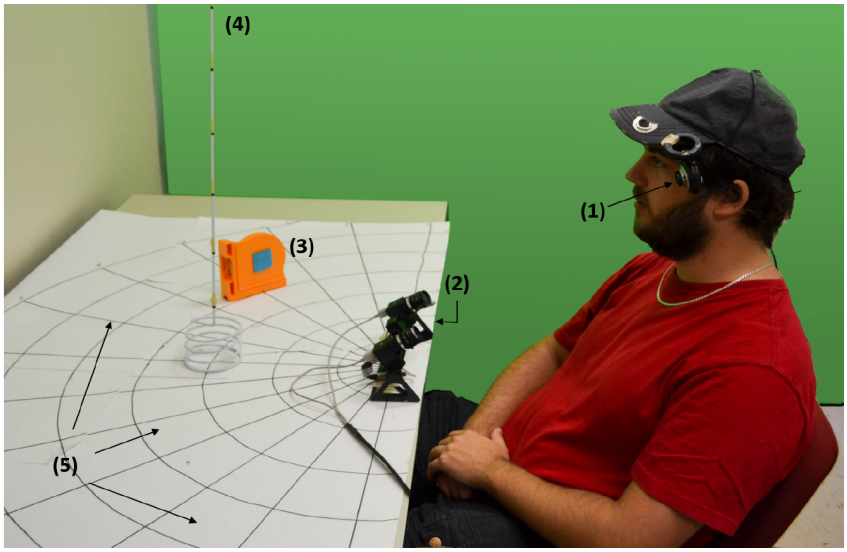

Fig. 2. Characterization setup: (1) Tecknet Webcam. (2)faceLAB 5 Eye tracker. (3) Target. (4) Vertical holder. (5) Cylindrical grid. workspace indicator

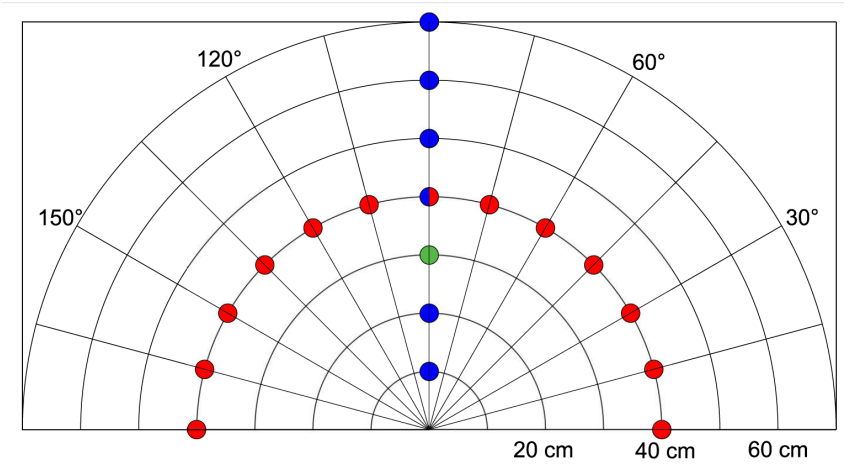

Fig. 3. Top view of the cylindrical grid of targets seen by the user. Red: angular series. Blue: radial series. Green: height and time series

A total of 33 healthy adults volunteered for the experiment. The mean (μ), median and standard deviation (σ) of the TRE for all 33 subjects for each point for both ETCV and the faceLAB 5 estimation of the 3DPOG.

3. RESULTS

Using the procedure described in the previous section, the position of a target in a 3D workspace and then the TRE were computed. For comparison purposes (3DPOG vs. ETCV), the TRE obtained with the estimation of the crossing point of both eye gaze direction, i.e. the 3DPOG, was also calculated.

During the measurements of the TRE against the angular position, for the targets located on each extreme side, at 0°, 15°and 180°, the subjects had to turn their head

to look directly at the target. For some subjects, their nose, another physiological trait or even the webcam would come between their eye and the eye tracker. This leads to a high proportion of the eye tracking sample to be rejected by the data filtering and thus leads to much higher TRE and variability on those critical points. The same holds for the highest points of the workspace, where either the cheeks or the eyelids of some subjects will hide the pupil from the eye tracker. These critical points are rendered in color later in Fig. 12.

Following are illustrated the TRE averaged from the 33 subjects (μ) and the standard deviation (σ). The median of the distribution is also drawn to illustrate the behavior of the error on the critical points for the subjects for whom no problem occurs. Finally, the best fitting curve of the median is also displayed. The TRE values for each angular, radial and vertical point are illustrated in Fig. 4, 5 and 6 respectively. The acquisition time dependency of the TRE is illustrated in Fig. 7.

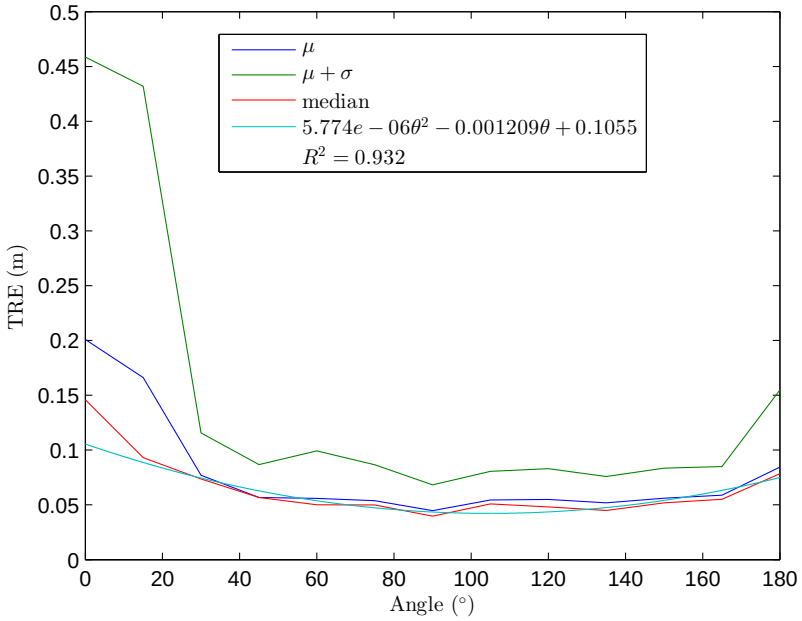

Fig. 4. TRE vs angular position of the target

To compare this data (3DPOG vs ETCV), Figs. 8-11 illustrates the TRE obtained using the point of gaze estimation of faceLAB 5 over the same points. It is worth noting that the TRE in Figs. 8-11 is one order of magnitude larger than the TRE in Figs. 4-7.

Fig. 12 shows a 3D rendering of the TRE over the workspace under study. Each sphere has a radius corresponding to the mean TRE at the corresponding location. For a clearer rendering, previously mentioned critical points were only rendered considering subjects with less than 80 % of the eye tracking sample filtered out. The color code indicates the number of subjects ignored for the corresponding point.

Table I summarizes the average comparisons over the whole workspace between the 3DPOG and ETCV methods relatively to their mean distance ($\bar{x}$) from the subject, using the data presented on Figs. 4-11.

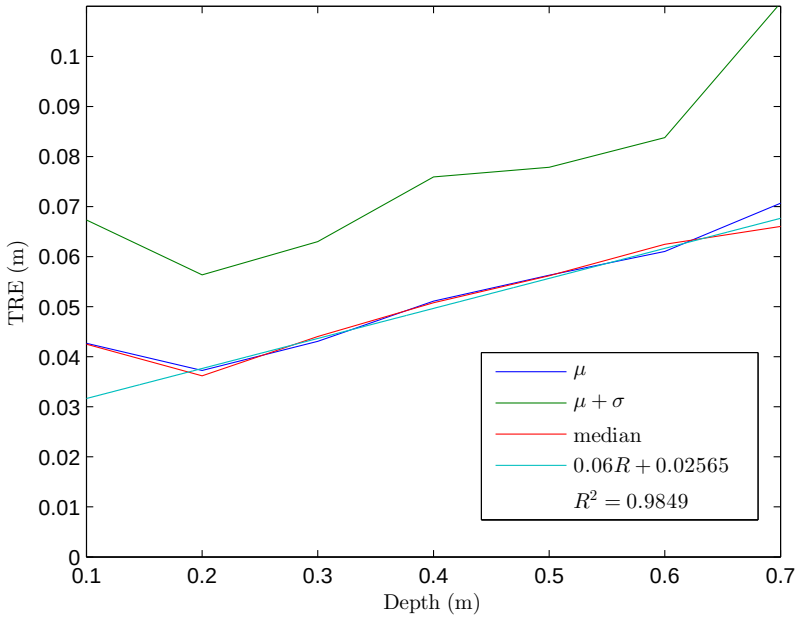

Fig. 5. TRE vs depth of the target

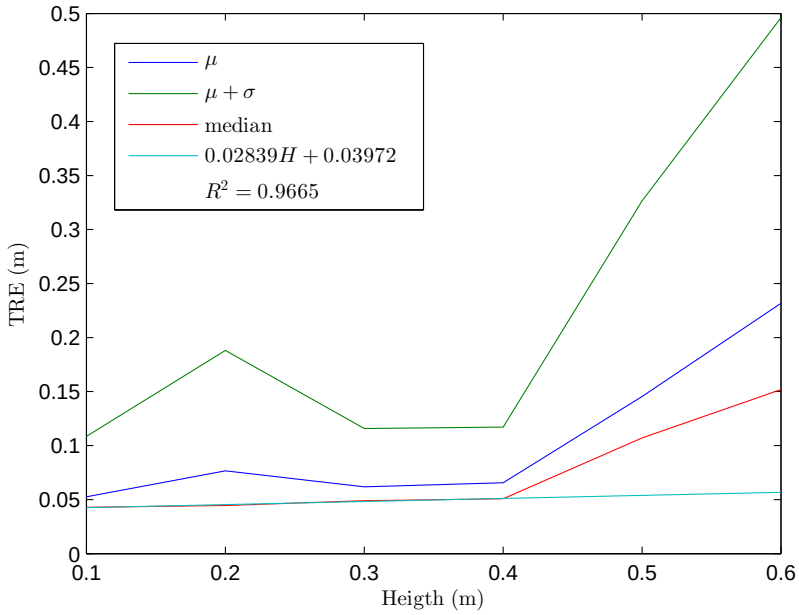

Fig. 6. TRE vs height of the target

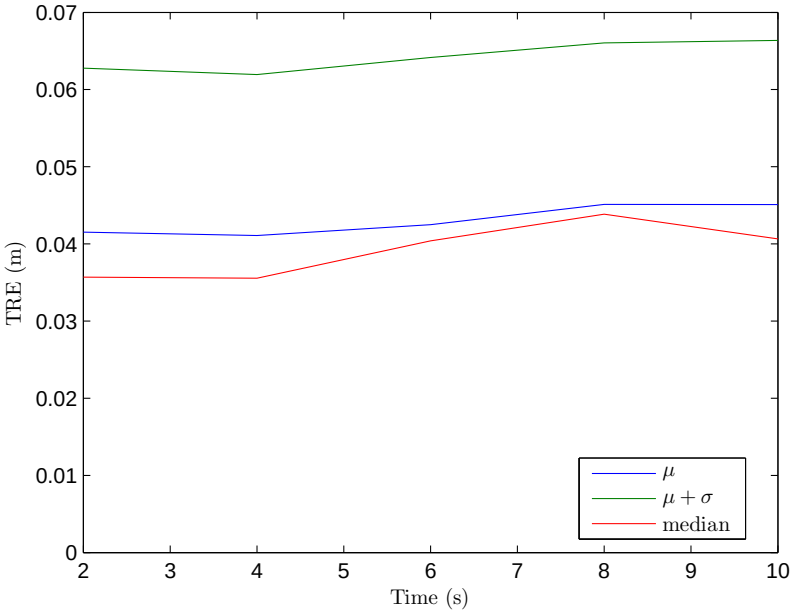

Fig. 7. TRE vs Acquisition time

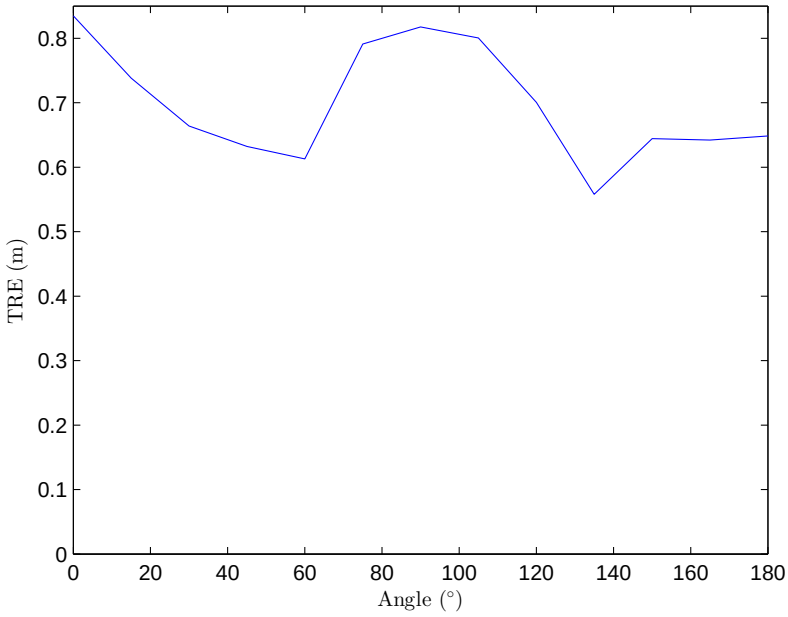

Fig. 8. TRE vs angular position of the target

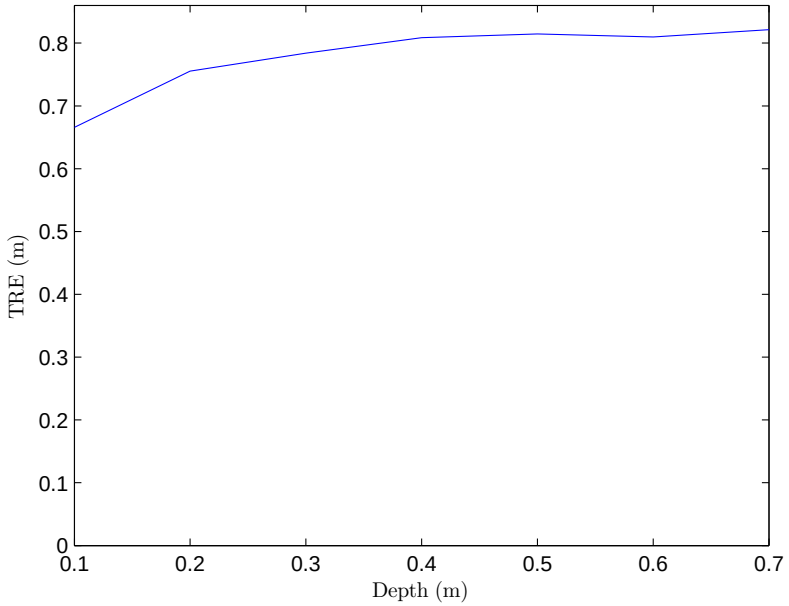

Fig. 9. TRE vs depth of the target

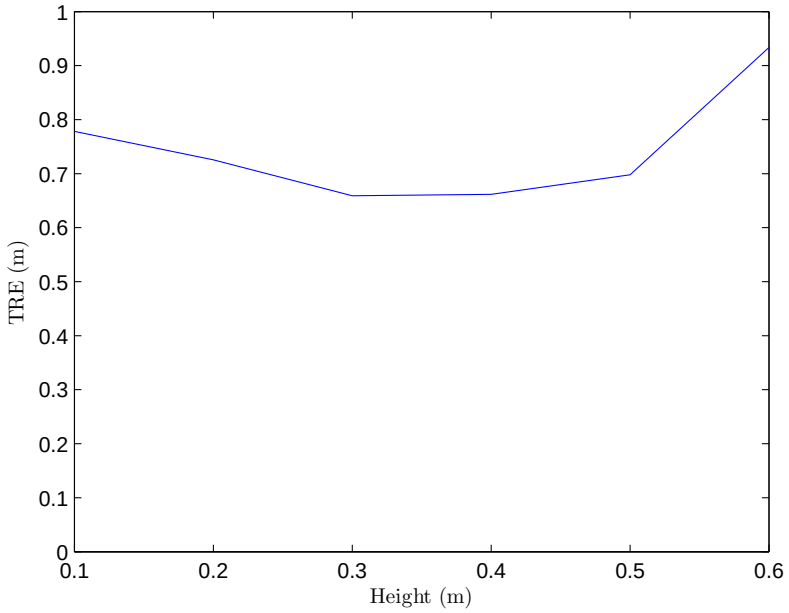

Fig. 10. TRE vs height of the target

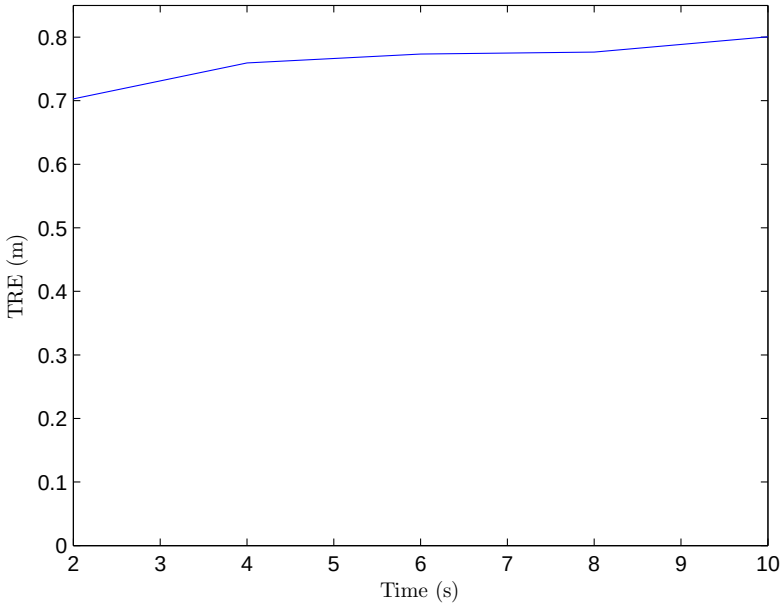

Fig. 11. TRE vs Acquisition time

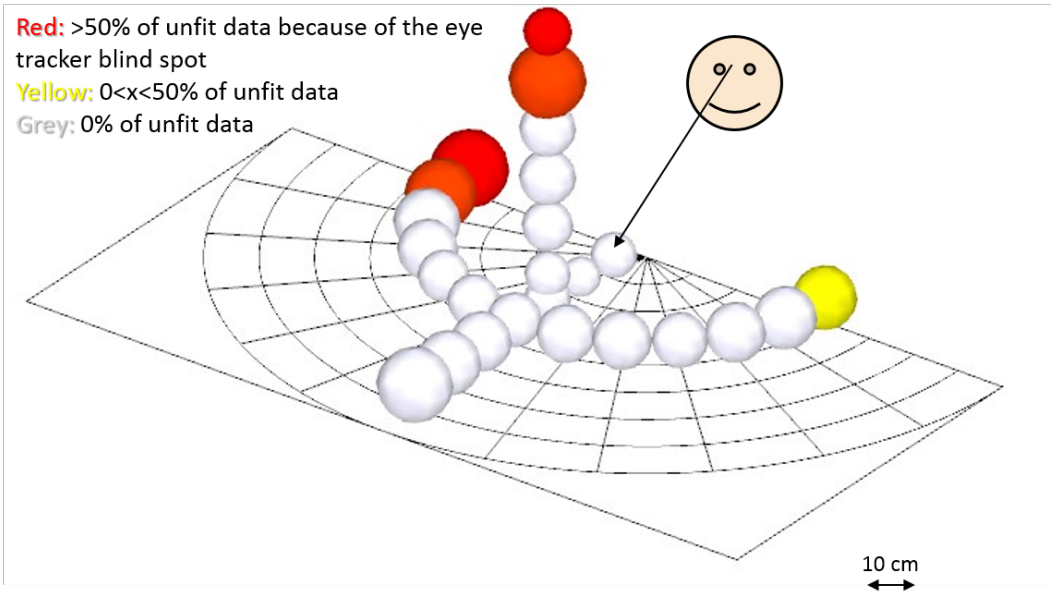

Fig. 12. Mean TRE and unacceptable data over a semi-cylindrical workspace of radius 70cm and height 60 cm

4. DISCUSSION

4.1. Results analysis

The objective of this paper was to characterize the accuracy and workspace of the method presented in [Leroux et al. 2015]. We consider as benchmarks the methods

Table I. Overall comparison between the ETCV results and the 3DPOG data acquired by faceLAB 5

Position set	$\bar{x}$	$\frac{\mu_{TRE(3DPOG)}}{\mu_{TRE(ETCV)}}$
Angular	0.89	10.82
Radial	0.74	15.6
Vertical	0.80	8.97
Acquisition time	0.94	17.72
GLOBAL	**0.84**	**12.65**

presented in [Abbott and Faisal 2012] which has a workspace of 60 cm x 50 cm x 40 cm located 50 cm away from the subject and in [Hennessey and Lawrence 2009] which uses a workspace of 30 cm x 23 cm x 25 cm about 25 cm away from the subject. As mentioned earlier, to the best of our knowledge, no other study describe the workspace considered while assessing the accuracy of a 3DPOG method, thus leaving no more possibilities for comparison.

As it should be expected, the TRE in Fig. 4 is minimal when the angular position of the target is right in front of the subject and the general behavior of the data is symmetrical around this point. The best fitting polynomial does not have a minimum at 90°but this is likely a numerical aberration brought by the abnormal increase in the error at the critical points discussed previously. Those points failed to be computed accurately because of the inability of the eye tracker to follow both eyes at such extreme positions for more than 50 % of our subjects. Although the increase in error is important, for many applications, this situation either does not appear or can be avoided easily by moving the eye tracker with the user, for example by using a head mounted eye tracking device such as in [Noris et al. 2011]. Outside of the critical points, the standard deviation of the error is very stable at 3.0 cm.

The linear increase of the TRE with depth in Fig. 5 can be easily explained by the increased absolute error generated by the constant angular error intrinsic to the system. The increase of the error not following this behavior at 0.1 m is caused by the image analysis part of the data processing, because the target is so close to the origin that the eye tracker itself obstructs part of the line of sight of the webcam. Again, the standard deviation is stable over the workspace under study, this time around 2.5 cm.

Given that the distance between the subject and the target is shortened when the height of the target increases, we expected a negative linear dependency of the TRE with the height in Fig. 6. Although, because of the critical points at 0.5 and 0.6 m high where the eye tracking data does not follow the same behavior, as discussed earlier both in this section and in the Results section, only 4 data points were used for the computation of the best fitting curve, therefore our result might not be significant in this particular case. More than 50 % of our subjects had more than 80 % of their eye tracking sample filtered out, thus leading to the very large TREs. Contrary to the previous data, the standard deviation of the TRE seems very variable, ranging from 5.2 cm to 26.4 cm.

The idea behind characterizing the TRE relatively to the acquisition time in Fig. 7 was to see the importance of the human capacity to stare at a single point in space. We expected to see the TRE decrease with increasing acquisition time because the larger eye tracking sample would better average out the random eye movement around the desired point. Our results show that the TRE is either constant or slightly increases linearly with acquisition time, our number of data points is too low for the difference to be significant. This result can be explained by two phenomena. Either the range

of acquisition time studied was not large enough to allow significant error averaging during the experiment, or as one would expect the focus of the subject is better for short acquisition time, compensating for the averaging with better accuracy. However, the exact behavior of the error over a larger range of acquisition time is of little importance since the potential applications of this method are most likely to be in real time. Our results indicate that an acquisition time of 2 seconds or maybe even less is sufficient to have proper eye tracking accuracy.

The faceLAB 5 software relies on the 3DPOG, which is, as mentioned earlier, widely used in the literature, to estimate the location of the target. However, our workspace is larger than the studies presented in [Abbott and Faisal 2012; Hennessey and Lawrence 2009] and no refinement of the method is implemented, so larger errors are to be expected. The calculation of the 3DPOG made by faceLAB 5 to estimate the target location did not show a robust behavior over the workspace, as seen in Figs. 8-11. The mean TRE using this method over the whole workspace is 73.6 cm. In [Leroux et al. 2015], an analytical derivation predicted an error of 72 cm when using the 3DPOG method at similar ranges. This error is way too large for any potential application, this is why eye tracking is often used with a screen or a mapping device when the viewed object needs to be located.

Using the combination of eye tracking and computer vision, our median error over the workspace ranges between 3.6 and 15.2 cm. If we exclude the critical points colored in Fig. 12, the maximum median error becomes 7.4 cm. Thus, in the main part of the workspace, our method is more than 12.65 times more accurate than the calculation of the point of gaze (see Table I). This kind of accuracy is sufficient for many potential applications, especially considering the fact that, in practical applications, the target will allow a certain region of tolerance, which may be of several millimetres, while we calculated our error relatively to the exact center of our target.

The ECTV method for calculating the location of the target has similar error as those described in [Abbott and Faisal 2012] and [Hennessey and Lawrence 2009], which had mean TRE of 5.8 and 3.93 cm and standard deviations of 4.7 and 2.8 cm respectively. Both used a refined version of the calculation of the 3D point of gaze. However, the workspace under test in both those papers are much smaller and the targets are closer to the subject ($50cm$ and $25cm$), thus leading intrinsically to smaller TREs. Moreover, neither of them studied the behavior of the error throughout their workspace. They also make use of custom made eye tracker systems and software, while this method is compatible with any commercially available eye tracker and can be implemented with only rudimentary knowledge of eye tracking.

A few more advantages come from using the ETCV method. Data relative to the electric and computing powers of our system was not acquired during our characterization experiment. However, it seems self-evident that image acquisition hardware consisting of a single webcam will take less resources than a stereovision system. Also, in stereovision systems, image processing has to be done twice, and further processing is still necessary to combine the data from each image, while ETCV requires to process only a single image. Therefore the ECTV method can be considered more appropriate for real-time and portable applications.

4.2. Limitations

4.2.1. Limitations of the ETCV method. The faceLAB 5 eye tracker has an intrinsic tracking rotation error of $\pm 1^\circ$ [Seeing Machines 2009]. This generates a TRE that will increase with the distance to the target. This error cannot be avoided and could only be improved by the use of a better eye tracker. However, faceLAB 5 has an accuracy similar to most other image-based eye trackers on the market [INC 2011; Morimoto and

Mimica 2005], so unless new devices are designed with significantly higher precision, this error source cannot be reduced.

More error is added by the computation of the radial coordinate from the picture taken by the webcam. The overall contribution to the TRE related to this matter is however impossible to quantify analytically. The discretization of the picture into pixels adds an upper bound to the accuracy attainable for the radial coordinate of the target. A camera with a higher resolution would generate better results, but also increase the computing time required, so this solution cannot be implemented in any application. Moreover, image processing, such as filtering and image erosion also distort the image. Finally, the target detection routine uses a color recognition algorithm, chosen because of the simplicity of its implementation, but more efficient and powerful algorithms could be used [Andreopoulos and Tsotsos 2013], for example from the OpenCV library [Itseez 2015].

The working principle of the ETCV method requires prior knowledge of the targets in order to compute their location. This means that our system would fail if set to work in a completely new environment. However, it will work correctly in a semi-controlled environment where all potential targets are either pre-characterized for their size and color or marked with a standardized marker.

4.2.2. Limitations of the characterization protocol. We chose to use the Tecknet camera for its very low price, relatively high resolution and its light weight and size. Although our results reached the targeted accuracy, this choice led to minor drawbacks that slightly increase TRE, which would be avoidable in future implementations with another camera. First, even if it was small, the camera was still too large to fit between the eyes of the subject without obstructing his/her field of view. The camera had to be placed on the side of the head, creating a parallax error. Also, this particular camera appeared to be sensible to the IR laser of the eye tracker. The picture would then be saturated lest it was taken at a separate time from the eye tracking data. This time separation may cause mismatches between the head position at the time of the eye tracking sampling and the acquisition of the field of view.

The curve fitting characterizing the behavior of the TRE over the space was each time done with only a handful of points, 13, 7 and 4 points for the angular, radial and vertical positions respectively. The distribution of the points over the space allows for a good general visualisation of the behavior of the accuracy, but more points would be required to have confidence in the exact quantitative results. Still, since the behavior of the TRE (i.e. the curve fitting results) is mostly useful to identify the origin of the imprecision, statistically robust results are arguably not necessary for practical applications.

5. CONCLUSION

In this paper, our objective was to assert the accuracy of the ETCV method over a large workspace. We showed that the calculation of the 3D point of gaze did not give sufficient precision for most practical applications where the user is not very close to his or her target. By using the eye tracker only for angular directions, as it is designed for, and adding a computer vision system with a webcam, the ETCV method was able to locate a target in 3D space with a TRE more than 90% inferior to the TRE of the 3D point of gaze calculation method (see Table I). Thirty three subjects used the system to locate targets in a cylindrical workspace. This data was used to characterize the behavior of the accuracy of the system over the workspace. This highlighted critical points on the edge of the workspace where the eye trackers sight is limited. Elsewhere, we obtained a median TRE ranging between 3.6 and 7.4 cm (see Fig. 4-7). The accuracy could still be improved with some changes in the equipment used for the data acquisition process

and with better performing algorithms for the image analysis. With the current accuracy, many potential applications become possible, such as visually controlled robotic assistants in the field of rehabilitation and adaptation engineering.

ACKNOWLEDGMENTS

The authors would like to thank the FRQNT/INTER and NSERC/BRPC for their financial support. Further, we are grateful to Pamela Didier and Thierry Adadja for their help during the characterization steps of the project.

REFERENCES

WW Abbott and AA Faisal. 2012. Ultra-low-cost 3D gaze estimation: an intuitive high information throughput compliment to direct brain–machine interfaces. *Journal of neural engineering* 9, 4 (2012), 046016.

Concetta F Alberti, Luciano Gamberini, Anna Spagnolli, Diego Varotto, and Luca Semenzato. 2012. Using an eye-tracker to assess the effectiveness of a three-dimensional riding simulator in increasing hazard perception. *Cyberpsychology, Behavior, and Social Networking* 15, 5 (2012), 274–276.

Michael Riis Andersen, Thomas Jensen, Pavel Lisouski, Anders Krogh Mortensen, Mikkel Kragh Hansen, Torben Gregersen, and Peter Ahrendt. 2012. *Kinect depth sensor evaluation for computer vision applications*. Technical Report. Aarhus University, Department of Engineering.

Alexander Andreopoulos and John K Tsotsos. 2013. 50 Years of object recognition: Directions forward. *Computer Vision and Image Understanding* 117, 8 (2013), 827–891.

Rowel Atienza and Alexander Zelinsky. 2003. Interactive skills using active gaze tracking. In *Proceedings of the 5th international conference on Multimodal interfaces*. ACM, 188–195.

Rowel Atienza and Alexander Zelinsky. 2005. Intuitive human-robot interaction through active 3d gaze tracking. In *Robotics Research. The Eleventh International Symposium*. Springer, 172–181.

Rositsa Bogdanova, Pierre Boulanger, and Bin Zheng. 2015. Three-Dimensional Eye Tracking in a Surgical Scenario. *Surgical innovation* (2015), 1553350615573581.

Andrew SA Chetwood, Ka-Wai Kwok, Loi-Wah Sun, George P Mylonas, James Clark, Ara Darzi, and Guang-Zhong Yang. 2012. Collaborative eye tracking: a potential training tool in laparoscopic surgery. *Surgical endoscopy* 26, 7 (2012), 2003–2009.

Andrew Duchowski, Eric Medlin, Nathan Cournia, Hunter Murphy, Anand Gramopadhye, Santosh Nair, Jeenal Vorah, and Brian Melloy. 2002. 3-D eye movement analysis. *Behavior Research Methods, Instruments, & Computers* 34, 4 (2002), 573–591.

Andrew T Duchowski. 2002. A breadth-first survey of eye-tracking applications. *Behavior Research Methods, Instruments, & Computers* 34, 4 (2002), 455–470.

Kai Essig, Marc Pomplun, and Helge Ritter. 2006. A neural network for 3D gaze recording with binocular eye trackers. *The International Journal of Parallel, Emergent and Distributed Systems* 21, 2 (2006), 79–95.

Antonio Frisoli, Claudio Loconsole, Daniele Leonardis, Filippo Banno, Michele Barsotti, Carmelo Chisari, and Massimo Bergamasco. 2012. A new gaze-BCI-driven control of an upper limb exoskeleton for rehabilitation in real-world tasks. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 42, 6 (2012), 1169–1179.

Hironobu Fujiyoshi, Yuto Goto, and Makoto Kimura. 2012. Inside-Out Camera for Acquiring 3D Gaze Points. In *IEEE Workshop on Egocentric (First-Person) Vision, CVPR*.

Craig Hennessey and Peter Lawrence. 2009. Noncontact binocular eye-gaze tracking for point-of-gaze estimation in three dimensions. *Biomedical Engineering, IEEE Transactions on* 56, 3 (2009), 790–799.

EYETRACKING INC. 2011. Hardware: Eye tracking systems. http://www.eyetracking.com/Hardware/Eye-Tracker-List. (2011). Accessed: 2016-01-15.

Itseez. 2015. Open source computer vision library. opencv.org/about.html. (2015). Accessed: 2016-01-15.

Muhammad Asghar Jan and Syed Majid Ali Shah Bukhari. 2009. Eye Tracking Interface Design for Controlling Mobile Robot. *SWAET 2012* (2009), 48.

M. Leroux, M. Raison, T. Adadja, and S. Achiche. 2015. Combination of eyetracking and computer vision for robotics control. In *Technologies for Practical Robot Applications (TePRA), 2015 IEEE International Conference on*. 1–6. DOI:http://dx.doi.org/10.1109/TePRA.2015.7219692

Robert Gabriel Lupu and Florina Ungureanu. 2013. A survey of eye tracking methods and applications. *Bul Inst Polit Iaşi* (2013), 71–86.

Päivi Majaranta and Andreas Bulling. 2014. Eye tracking and eye-based human–computer interaction. In *Advances in Physiological Computing*. Springer, 39–65.

Microsoft. 2016. Kinect for Windows v2. http://www.microsoft.com/en-us/kinectforwindows/meetkinect/features.aspx. (2016). Accessed: 2016-01-15.

Carlos H Morimoto and Marcio RM Mimica. 2005. Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding* 98, 1 (2005), 4–24.

Basilio Noris, Jean-Baptiste Keller, and Aude Billard. 2011. A wearable gaze tracking system for children in unconstrained environments. *Computer Vision and Image Understanding* 115, 4 (2011), 476–486.

Lucas Paletta, Katrin Santner, Gerald Fritz, Heinz Mayer, and Johann Schrammel. 2013. 3D Attention: Measurement of Visual Saliency Using Eye Tracking Glasses. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 199–204. DOI:http://dx.doi.org/10.1145/2468356.2468393

Jinglin Peng. 2011. *Comparison of three dimensional measurement accuracy using stereo vision*. Ph.D. Dissertation. Faculty of Graduate Studies and Research, University of Regina.

Thies Pfeiffer, Marc Erich Latoschik, and Ipke Wachsmuth. 2008. Evaluation of binocular eye trackers and algorithms for 3D gaze interaction in virtual reality environments. *JVRB - Journal of Virtual Reality and Broadcasting* 5, 16 (2008).

Stanislav Popelka, Alzbeta Brychtova, and Jan Brus. 2012. Advanced Map Optimalization Based on Eye-tracking. In *Cartography: A tool for spatial analysis*. 1–20.

Edward Rosbergen, Rik Pieters, and Michel Wedel. 1997. Visual attention to advertising: A segment-level analysis. *Journal of consumer research* 24, 3 (1997), 305–314.

E. Schneider, T. Dera, K. Bard, S. Bardins, G. Boening, and T. Brand. 2005. Eye movement driven head-mounted camera: it looks where the eyes look. In *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, Vol. 3. 2437–2442 Vol. 3. DOI:http://dx.doi.org/10.1109/ICSMC.2005.1571514

Seeing Machines 2009. *faceLAB 5: Eye tracking for research*. Seeing Machines, www.seeingmachies.com.

Shoichi Shimizu and Hironobu Fujiyoshi. 2011. Acquisition of 3D Gaze Information from Eyeball Movements Using Inside-out Camera. In *Proceedings of the 2Nd Augmented Human International Conference (AH '11)*. ACM, New York, NY, USA, Article 6, 7 pages. DOI:http://dx.doi.org/10.1145/1959826.1959832

Maria Staudte and Matthew Crocker. 2008. The utility of gaze in spoken human-robot interaction. In *Proceedings of Workshop on Metrics for Human-Robot Interaction 2008, March 12th*. Citeseer, 53–59.

J.N. van der Geest and M.A. Frens. 2002. Recording eye movements with video-oculography and scleral search coils: a direct comparison of two methods. *Journal of Neuroscience Methods* 114, 2 (2002), 185 – 195. DOI:http://dx.doi.org/10.1016/S0165-0270(01)00527-1

K. P. Weber, G. M. MacDougall, H. G.and Halmagyi, and Curthoys I. S. 2009. Impulsive Testing of Semicircular-Canal Function Using Video-oculography. *Annals of the New York Academy of Sciences* 1164 (2009), 486–491.

Zhefan Ye, Yin Li, Alireza Fathi, Yi Han, Agata Rozga, Gregory D. Abowd, and James M. Rehg. 2012. Detecting Eye Contact Using Wearable Eye-tracking Glasses. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp '12)*. ACM, New York, NY, USA, 699–704. DOI:http://dx.doi.org/10.1145/2370216.2370368