J. Biomed. Eng. Res.
Vol.26, No.3, 133-138, 2005

A New Ocular Torsion Measurement Method Using Iterative Optical Flow

InBum Lee[1], ByungHun Choi[1], SangSik Kim[2], Kwang Suk Park[3]

[1]Interdisciplinary Program of Medical and Biological Engineering Major, Seoul Nat'l Univ.
[2]SLMed, [3] Departments of Biomedical Engineering, College of Medicine, Seoul Nat'l Univ.
(Received December 29, 2004. Accepted March 28, 2005)

Abstract: This paper presents a new method for measuring ocular torsion using the optical flow. Images of the iris were cropped and transformed into rectangular images that were orientation invariant. Feature points of the iris region were selected from a reference and a target image, and the shift of each feature was calculated using the iterative Lucas-Kanade method. The feature points were selected according to the strength of the corners on the iris image. The accuracy of the algorithm was tested using printed eye images. In these images, torsion was measured with 0.15° precision. The proposed method shows robustness even with the gaze directional changes and pupillary reflex environment of real-time processing.

Key words: Ocular torsion, Optical flow

INTRODUCTION

In order to understand eye movements and ocular motor function, it is necessary to measure torsional eye movement, as well as horizontal and vertical movement. The most commonly used three-dimensional eye tracking methods are the sclera search coil and VOG using image processing. The sclera coil is an accurate and high temporal resolution device, but has the limitations of invasiveness to the subject, restricted testing time and problems with slippage.

Recent improvements in the performance of personal computers and image processing techniques have allowed the capture of real-time torsional measurements using the tracking of eye movements. The principle of torsional measurement by image processing is based on the calculation of the relative displacement of the iris signature by comparing features of a reference and target image. Polar cross correlation, the most simple method of measuring torsion [1][2], uses a one-dimensional iris signature and does not require a time consuming calculation process. However, its results are strongly affected by image quality. A high-level pattern recognition method using feature points on the iris has been suggested, but it is time consuming and is not appropriate for a real-time system [3][4][6].

There are several factors that make torsional measurements challenging. The iris image under infrared illumination is indistinctive, which makes tracking of features difficult. Geometric distortion is another issue of concern, in that the eye image projected to the image plane is deformed according to the rotation of the eyeball [1][2]. If relevant parameters are known, it is easy to compensate for the position of the features, but, in practice, it is difficult to obtain accurate parameters for each subject. Deformation of the iris signature caused by the pupillary reflex is another problematic aspect of torsional assessment. During measurements, the pupil can dilate or constrict in response to light stimulation or as a result of emotional changes and/or pharmaceuticals that affect the autonomic nervous system [12]. When the pupillary reflex occurs, the iris signature does not constrict regularly to a radius or an angle. Lastly, achieving adequate simplicity of processing is a concern. An acceptable method of measurement must use fairly simple calculations in order to function in a real-time system which should process over 30 frames per second. This paper presents a new method for measuring torsional movement which can overcome such difficulties by using optical flow.

This study was supported by a grant of the Korea Health 21 R&D Project, Ministry of Health & Welfare, Republic of Korea(01-PJ1-PG1-01CH16-0002)
Corresponding Author: Kwang Suk Park
Department of Biomedical Engineering, College of Medicine, Seoul National University
Tel. 02-760-3135 Fax. 02-3676-1175
E-mail. kspark@bmsil.snu.ac.kr

METHODS

An optical flow reflects image changes due to motion during a time interval [7]. The advantages of optical flow include simplicity of calculation and robustness of tracking. In the current method, optical flow was used to track the feature points on the iris, and the torsional angle was determined by the displacement of these features. The tracking procedures were applied to each feature independently. An iterative Lucas-Kanade method, which is computationally simple and powerful in tracking when the pixel displacement is small, was used to calculate the optical flows. Compared to the frame rate of the camera, the ocular torsion was slow enough that the assumption of a small displacement could be satisfied.

The process was divided into 4 parts: preprocessing, feature selection, feature tracking and ocular torsion measurement.

Preprocessing

The object of preprocessing was to extract the iris from the eye image and transform the circular shape of the iris into a rectangular image.

We began with the assumption that the location and size of the pupil were known. The tracking methods used have been explained in several earlier papers [8][9][10][11]. Using the pupil location and size, the pupil-centered image was cropped from the original image (Fig.1). In order to extract the iris region from the eye image, the appropriate inner and outer radii were selected, and the polar coordinates were converted into Cartesian coordinates (Fig.1). The inner radius represented the size of the pupil, and outer radius was determined by adding a constant offset to the inner radius. If the center of the pupil was set as the origin, then the coordinate of each point in the iris could be expressed in a polar coordinate system.. The radius and angle in polar coordinates determined the x- and y-values of the Cartesian coordinates of the iris image. Bilinear interpolation was used to obtain floating-point accuracy.

This process has two advantages in tracking features. First are the rotation invariant features. With rotation, the features are rotated with the eyeball and require additional processing in order to identify their orientation. Transforming the Cartesian coordinates from the polar coordinates results in features maintaining their orientation during torsion. The rotational movement of the iris signature is converted into horizontal movement. Second is the compensation for pupillary reflexes. Because the inner radius is determined by the pupil size, the transformed image always includes the iris, regardless of changes in the pupil radius. This prevents the features from leaving their position in a radial direction in response to pupil dilation and constriction.

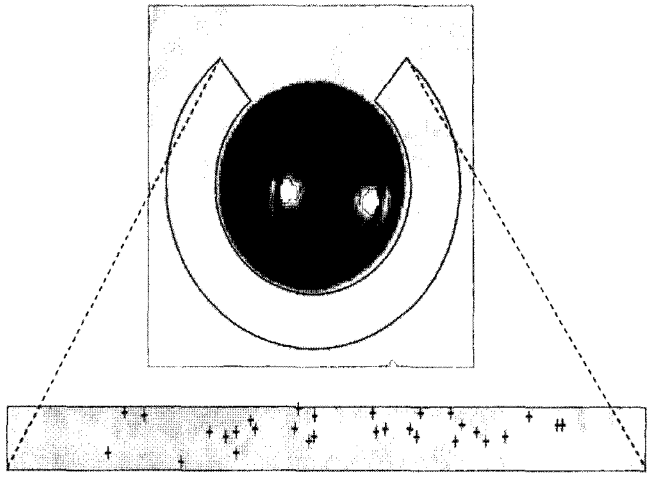


Fig.1. Pupil image cropped from the original image. The iris signature has been extracted from the pupil image and transformed into a rectangular image. Cross marks indicate selected feature points

Feature Selection

Better tracking performance can be achieved by selecting the appropriate features to track [5]. Such features are assessed according to the strength of the corner, which is evaluated with an eigenvalue of matrix C within a small window around the feature.

$$\mathbf{C}=\begin{bmatrix} \sum E_x^2 & \sum E_x E_y \\ \sum E_x E_y & \sum E_y^2 \end{bmatrix} \tag{1}$$

E_x and E_y are spatial gradients of images on the x- and y-axis. From matrix **C**, 2 eigenvalues and eigenvectors were calculated which encode the direction and strength of the edge components within the window. A large second eigenvalue guaranteed that there were 2 strong orthogonal edge components and the existence of an inverse matrix.

To calculate torsional angle, 30 feature points were selected within the iris image, according to the magnitude of the second eigenvalue (Fig.1). In order to eliminate the disturbance from the eyelashes and eyelid, a 45° area of each end side was excluded. Each feature point was chosen using the criterion of a minimum distance from other points. The selected feature points on the reference iris signature are shown in Fig.2. The feature points were selected just once, at the start of measurement, and were used as reference points throughout the whole measurement process.

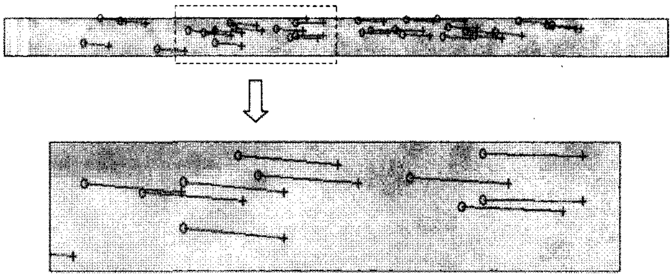


Fig.2. Feature tracking using an iterative optical flow. The circle is the initial position and the cross is the tracking result. The features were tracked along torsional and radial directions.

Feature Tracking

From the target image, an iris image of the same size as the reference image was extracted, and the displacement of the features between the reference and the target image was calculated. The selected features were tracked in the target image using an iterative Lucas-Kanade algorithm [7]. A brief description follows:

Let the reference and target image be set as image A and image B, respectively. The displacement vector (v_x, v_y) of the feature point (p_x, p_y) between 2 images A(x, y) and B(x, y) is defined within a window size of w_x around the feature. The goal is to find the displacement vector (v_x, v_y) that minimizes the cost function $\varepsilon(v)$:

$$\varepsilon(v) = \sum_{p_x-w_x}^{p_x+w_x} \sum_{p_y-w_y}^{p_y+w_y} (A(x,y) - B(x+v_x+g_x, y+v_y+g_y))^2 \tag{2}$$

The horizontal component of the displacement vector can be considered the torsional component of the iris. The variables (g_x, g_y) are initial estimates, which are set at 0 at the start. At the optimum, the first derivative of ε with respect to v is 0.

The optimum v can be calculated with the spatial and temporal gradient of the image [7].

$$v_{opt} = C^{-1} \cdot \overline{b} \tag{3}$$

where

$$\mathbf{C} = \sum_{p_x-w_x}^{p_x+w_x} \sum_{p_y-w_y}^{p_y+w_y} \begin{bmatrix} \sum E_x^2 & \sum E_x E_y \\ \sum E_x E_y & \sum E_y^2 \end{bmatrix}$$

$$\overline{b} = \sum_{p_x-w_x}^{p_x+w_x} \sum_{p_y-w_y}^{p_y+w_y} \begin{bmatrix} \delta E E_x \\ \delta E E_y \end{bmatrix}$$

Matrix **C** is calculated in the initial feature selection process, and E_x and E_y can be computed directly from the neighborhood of the feature points of the reference image. Therefore, a 2 x 2 matrix **C** is calculated just once and remains constant throughout the tracking process. This is a computational advantage given the limited resources in real-time processing. δE is the temporal derivative between the reference and the target image. The features were selected according to the magnitude of the second eigenvalue, which guarantees the existence of the inverse matrix of **C**.

In order to obtain an accurate solution, the general Lucas-Kanade process was iterated until sufficient precision had been obtained. The whole process was performed over an image pyramid for computational efficiency and a wider search region without a large window size. A pyramid level of 1 or 2 was sufficient. Each feature point was tracked independently.

The Lucas-Kanade algorithm works only when the displacement of the feature point is small. In order to maintain the proximity between the feature points in the reference and target images, the initial estimate points (g_x, g_y) were set to the torsional angle of the previous frame. Because the iris signature moves horizontally in a mass, the torsional angle of the previous frame could be used as the initial estimate point, even in cases in which some features were missed.

The results of the feature point tracking are shown in Fig.2. In the figure, the feature points are shown to move horizontally compared to the reference image.

Compensation

When the eye is projected on the image plane, the relative location between feature points and the pupil center changes. The torsional angle is determined according to the change in gaze direction, even though there is no torsional component because the surface of the eye is not parallel to the image plane. For accurate measurements of torsion, this geometric distortion should be taken into account. Previous papers suggested a compensation algorithm [1][2], and we applied the same geometric calculation [13]. The fundamentals are similar in spirit, but each feature point is compensated independently because they do not lie on a concentric circle.

When the visual axis lies in the same axis as the camera, the coordinates of the feature point P, projected on the image plane, are $(r_1 \cos\theta_1, r_1 \sin\theta_1)$ originated from the pupil center (Fig.3). By horizontal and vertical rotational matrix $R(\theta,\phi)$, the location of point P is transformed to $(r_2 \cdot \cos\theta_2, r_2 \sin\theta_2)$. The variables r_2 and θ_2 can be calculated with $R(\theta,\phi)$ with eyeball and camera parameters [14]. The torsional error ε of the feature point P is $\theta_1 - \theta_2$. Each feature point has its own torsional error, and a compensated angle can be computed by subtracting this value from the measurement. Our focus in this paper was on the investigation of algorithms for torsional measurements, so we were not concerned with the alignment of the visual axis and camera axis.

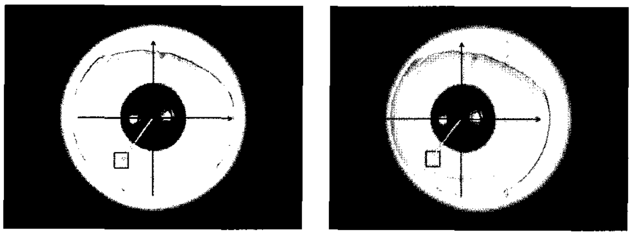


Fig. 3. Torsional error caused by image distortion. Feature location can be expressed as $(r_1 \cos\theta_1, r_1 \sin\theta_1)$ in (a). r_1 is the length of the white line, and θ_1 is the angle between the line and the x-axis. In (b), feature location is $(r_2 \cdot \cos\theta_2, r_2 \sin\theta_2)$. Because of image distortion, there is a difference between θ_1 and θ_2.

Determination of Torsional Angle

From one frame, 30 optical flows were obtained, some of which had an error due to occlusion by the eyelash, a reflection of illumination at the cornea or a step out from the image. In order to eliminate incorrect features, the feature points were classified into several groups according to the displacement. The group that was apart from the others was removed. The torsional angle was calculated as a mean of the displacement of features.

RESULTS

A three-dimensional eye tracking system was developed to test the presented method (Fig.4). A CCD camera was attached to a pair of water goggles, and 2 infrared LEDs for illumination and 4 white LEDs for light stimulus were attached to the inside of the goggles. Images from the camera were digitized using a frame-grabber program and processed using a personal computer. The processing software was developed using Visual C++ 6.0. A Pentium □ 3.0GHz CPU was used.

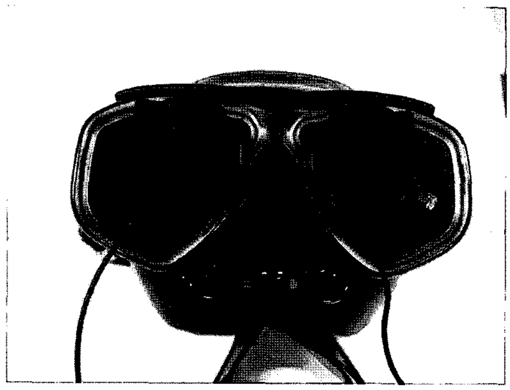


Fig. 4. Goggles to capture eye movements. Both eyes were captured simultaneously. Two infrared LEDs for illumination and 4 white LEDs for light stimulus were attached inside the goggles. Images from the camera were transmitted to a personal computer and processed in real-time.

Horizontal, vertical and torsional eye movements and pupil size were calculated. For each frame of the eye image, preprocessing, which extracts a portion of the iris with the position and radius of the pupil, took less than 4ms. In the extracted iris image, 30 features were tracked independently, and this procedure took less than 2ms. The tracking time depended upon the number of features and the number of iterations, but the suggested criteria were sufficient to measure torsion. All the images were processed in real-time, including finding the location and the size of the pupil.

To evaluate the performance of the presented method, we conducted experiments using printed eye images. The printed eye image experiment simulated the torsion measurement with the visual axis lying on the camera axis. The eye image was captured using a CCD camera. Image quality was then enhanced by adjusting the contrast of the pupil and iris regions. The enhanced image was printed with a gray scale printer. The apparatus for the experiment included a precise step motor and a gray scale CCD camera with infra-red LED. The step motor was a VEXTA UHG502B with 0.0072°/step precision, and the CCD was a standard NTSC camera (Fig.5). The printed eye picture was attached to the step motor which was controlled by a personal computer. The attached picture was rotated from 0° to 10° with 1.0008° steps over 35s, and the picture was captured using a CCD camera. Captured images were then analyzed using a personal computer in real-time. The result is plotted in Fig.6. The maximum error was less than 0.15°, and mean error was 0.0265°. The result shows that, if torsion is measured while the visual axis agrees with the camera axis, the error of torsional measurement is sufficiently small.

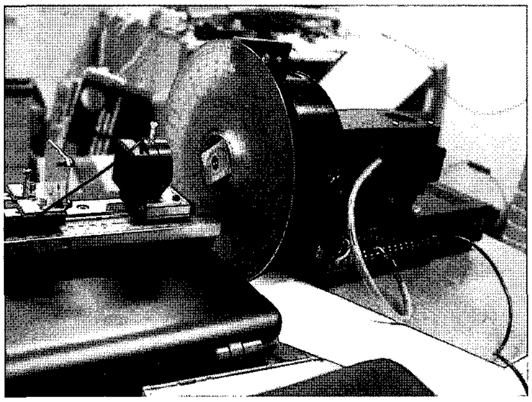
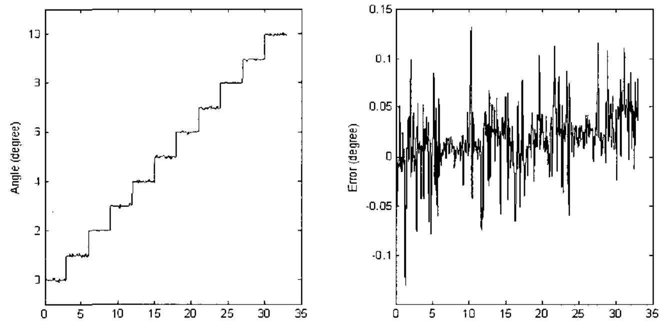


Fig. 5. Experimental setup for validation. Printed eye images were attached to the step motor which was controlled by a personal computer through a parallel port.



(a) (b)

Fig. 6. Results of the validation experiment using a step motor. (a) is the result of torsional measurements. Eye images were rotated from 0° to 10°. Errors in each frame are shown in (b). The result shows that the maximum error is less than 0.15°.

The robustness of torsional measurements in the presence of the pupillary reflex was also tested. Subjects wore the goggles with a CCD camera and adapted to darkness in order to dilate the pupils. Reference images were captured from the dilated images in the upright position, and feature points were selected. Then subjects were requested to roll their head right and left at 0.5Hz intervals, tuned with a beeping sound. In the middle of the head roll, the white LEDs inside the goggles were turned on and the pupils constricted. Fig.7 shows the radius of pupil and torsional measurements. In our results, the measurement of torsion was not affected by changes in pupil radius.

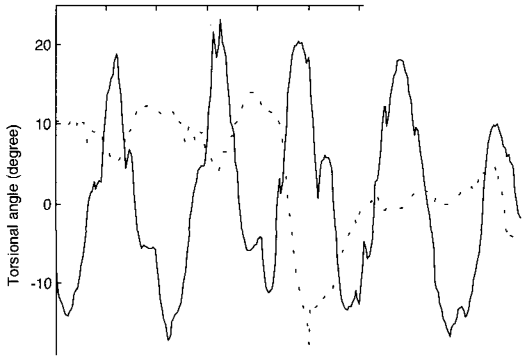


Fig. 7. Plot of torsional eye movement (solid line) and change in the pupil radius (dotted line). Though the pupil radius was reduced by light stimulus, torsional measurement was not affected.

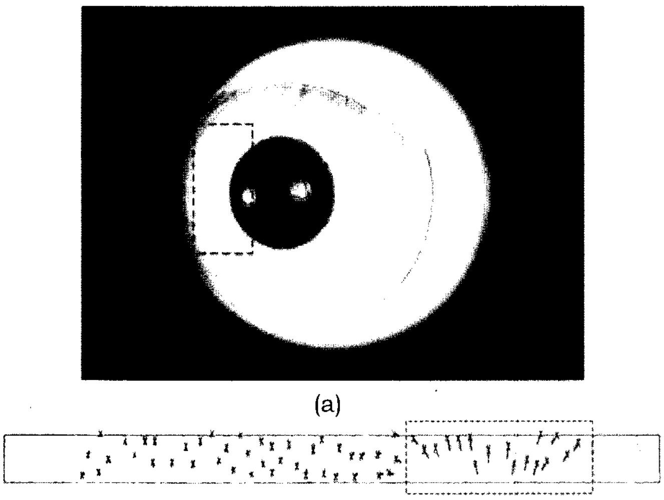
DISCUSSION

Computational simplicity is essential for a real-time system. In standard CCD cameras, the frame rate is 30 frames per second. This means that 33ms of processing time is allowed for each frame, so 6ms processing time per frame is sufficient to make a real-time system. Developed eye tracking systems work well in real-time. Although the location and size of the pupil needs to be known, this does not require a large amount of computation time.

The simplicity required of a real-time system is provided by the optical flow method because each feature is tracked based on a gradient of intensity between frames. Only the spatial and temporal gradient of a small region around the feature needs to be known.

Another significant advantage of this method is that each feature is tracked separately. Independence in movement of features allows the method to deal with irregular distortions. Fig.8 shows how the method works in the presence of geometrical compensation. In the dashed box of the iris image, tracking for the distortion is shown at each feature. This results in robust torsional measurements.

The independence of features is useful from the perspective of pupillary reflex as well. Changes in pupil radius are an important cause of error during torsion measurements. In the suggested method, two processes compensate for the pupillary reflex. One compensation takes place during the preprocessing stage, when the iris region is transformed into a rectangular shape, and the iris is always extracted from the pupil boundary. The second compensation involves the use of optical flow. Irregular movements of the iris signature may occur in the whole iris region during dilation and constriction of the pupil. This pupillary movement should be tracked in the radial direction, and this irregularity requires independent tracking of each feature. Optical flow offers an appropriate solution to this problem in that the features are tracked in the radial direction as well as the torsional direction. This prevents the method from being interrupted by geometric distortion and pupillary reflexes.



(a)

(b)

Fig. 8. Independence of feature movement. (a) The box indicates the area where image distortion occurred. (b) The cross marks indicate the tracking results, and the tail represents the shift from reference. Each feature was tracked independently.

CONCLUSION

This paper suggests a new method for obtaining ocular torsion measurements using optical flows. The optical flow method was used to track features in a cropped image of the iris around the pupil. An iterative Lucas-Kanade method was used to calculate the optical flow. The optical flow offers several advantages in measuring ocular torsion. It is simple enough for a real-time system, and the geometric distortion and pupillary reflexes can be compensated for by tracking features independently. It is believed that this method can be applied to a binocular real-time measurement system in clinical research.

REFERENCES

[1] T. Haslwanter, and S.T. Moore, *"A theoretical analysis of three-dimensional eye position measurement using polar cross-correlation"* , IEEE Trans. on Biomedical Eng., Vol. 42 , pp.1053-1061, 1995.

[2] S.T. Moore, T. Haslwanter, I.S. Curthoys, and S.T. Smith, *"Measurement of three dimensional eye position using image processing: a geometric approach Image Processing"*, Proceedings. ICIP-94., IEEE International Conference, Vol. 1 , pp. 436- 440, 13-16, 1994.

[3] D.W.J. Chung, M. Eizenman, B.S.K. Cheung, and R.C. Frecker, *"Estimation of ocular torsion with dynamic changes in pupil size"*, Engineering in Medicine and Biology Society, Engineering Advances, Vol.2, pp. 924 – 925, 1994.

[4] E. Groen, and *J.E.* Bos, *"Determination of ocular torsion by means of automatic pattern recognition"*, IEEE Trans. on Biomedical Eng., Vol. 43, pp. 471 – 479, 1996.

[5] Jianbo Shi, and Carlo Tomasi, *"Good features to track"*, Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn., pp.593-600, 1994.

[6] J.P. Ivins, J. Porrill, Frisby, J.P., *"Deformable model of the human iris for measuring ocular torsion from video images Vision"*, Image and Signal Processing, IEEE Proceedings, Vol. 145 , No. 3, pp.213 – 220, 1998.

[7] B. Lucas, and T. Kanade, *"An Iterative Image Registration Technique with an Application to Stereo Vision"* , Proc. of 7th International Joint Conference on Artificial Intelligence (IJCAI), pp. 674-679, 1981.

[8] Ying-li Tian, T. Kanade, and J.F. Cohn, *"Dual-state parametric eye tracking"* , Automatic Face and Gesture Recognition, Proceedings. Fourth IEEE International Conference, pp.110 – 115, 28-30, 2000.

[9] T.E. Hutchinson, K.P. Jr. White, W.N. Martin, K.C. Reichert, and L.A. Frey, *"Human-computer interaction using eye-gaze input"*, IEEE Transactions on Systems, Man and Cybernetics, , Vol. 19, No.6 , pp.1527 – 1534, 1989.

[10] Y. Ebisawa, *"Improved video-based eye-gaze detection method"*, IEEE Transactions on Instrumentation and Measurement, Vol. 47 , No. 4 , pp. 948 – 955, 1998.

[11] Xangdong Xie, R. Sudhakar, and Hanqi Zhuang, *"Real-time eye feature tracking from a video image sequence using Kalman filter"*, IEEE Transactions on Systems, Man and Cybernetics, Vol. 25, No. 12, pp.1568 – 1577, 1995.

[12] M.A. Phillips, P. Bitsios, E. Szabadi, et al., *"Comparison of the antidepressants reboxetine, fluvoxamine and amitriptyline upon spontaneous pupillary fluctuations in healthy human volunteers"* , Psychopharmacology (Berl), Vol.149, No.1, pp.72–76, 2000.

[13] T. Haslwanter, *"Mathematics of Three-dimensional Eye Rotations"*, Vision Res., Vol.35, pp.1727-1739, 1995.

[14] K. Schreiber, T, Haslwanter, *"Improving calibration of 3-D video oculography systems"*, IEEE Trans. on Biomedical Eng., Vol. 51, pp. 676- 679, 2004.