

Event driven Camera based Eye Tracking

MitraTajrobehkar and Mehran Jahed

1Department of Electrical Engineering, Sharif University of technology international campus, Kish Island, Iran

2Department of Electrical Engineering, Sharif University of technology, Tehran, Iran

ABSTRACT

This study proposes a general diagnostic approach based on a camera based eye-tracking technique and a discrete Kalman filter as an iris tracker. In an event driven scenario, the proposed iris-tracking method is used to locate and track the subject's eye movement and gesture by comparing it to a set of classified normal and typical gestures. The database is based on available videos as well as a number of experiments conducted with normal subjects. Proposed approach Utilized feature based methods of face, such as skin color while detection of the eyes utilized a histogram-based approach and SVM was used as a two-class classifier to divide region into eyes and non- eyes patterns. Based on the results, the proposed approach provide an efficient eye detection and tracking method with an average of 99.1% accuracy.

KEYWORD

Eye detection, Iris tracking, Kalman filter.

1. INTRODUCTION

Eye tracking research has been studied for many application and areas. In general, tracking of the eye has been done in two categories; intrusive, and non-intrusive. The intrusive technique employs devices such as electrodes, contact lenses, and head mounted. One of the challenging areas in this field concerns detection of abnormal movements of the iris which may provide reasonable clues for diagnostic of possible disorders.

Autism spectrum disorder concerns a group of complex disorder of brain development. A recent estimate (March 2014) states that one in every 68 children is born with ASD¹. Moreover, the number of children identified with autism continues to rise. It is said that an approach to tackle this problem is to diagnose the disorder before the age of three, as it can be better treated and it may minimize disease progression. This study proposes an approach that in effect can be utilized in such diagnostic approaches, being based on eye movement tracking and gaze direction.

To achieve this approach, recorded videos of normal subjects were utilized in such a manner to mimic normal and abnormal behaviour as stated in the literature. Briefly, since these videos capture the subject as well as the surroundings, the following steps were to be followed.

- Detection of face, eyes, and irises separately
- Tracking of irises from frame to frame
- Analysis of the tracking efforts in order to recognize the behavior

This paper consists of five sections. Section two provides a literature review. Section three depicts the theoretical background while section 4 presents the proposed methodology for effective detection and tracking of the iris in an effort to diagnose possible abnormal behavior, and finally in section 5 results and in section 6 the conclusion are provided

2. RELATED WORKS

In this section previous efforts to develop automatic face, eye and iris detection is introduced and briefly discussed

As noted before, face detection is a first step in this work. In general face detection can be classified in four approaches:

- Knowledge based method
- Feature based method
- Method on template matching
- Appearance based method

In what follows our main focus regarding face detection is and established literature is provided.

In 2001, Viola and Jones proposed a frontal face detection system within gray-scale images and based on the Adaboost learning algorithm [8]. (More details about this algorithm will be provided in section 3.)

In a separate work, a hybrid method utilizing colored images was later introduced [9]. The Haar feature-based face detector developed by Viola and Jones, which had been designed for gray-scale images, was combined with a skin-color filter, which provided complementary information in color images. The false detection of Haar-Feature based detector was eliminated using the skin color through post-filtering methods. The method achieved an accuracy of 95.75% for detection of face images utilizing the BAO database.

Later on in a more robust approach, a new Support Vector machine (SVM) classifier with a Gaussian kernel that detects eyes in grayscale, was introduced [26]. As expected In general, algorithms of eye detection may be categorized into active infrared-based and passive Image-based approaches. [18] In [13] a method of eyes detection in facial images using Zernike moment with Support Vector Machine (SVM), is introduced. There, the eye and non-eyes patterns are represented in terms of the magnitude of the Zernike moment and classified by the SVM. The method achieves matching rates of 94.6% for detection of eyes in face images from the ORL human face database [25].

In [25] a method for automatic detection of eyes in images of human faces using semi-variogram functions and support vector machines, is proposed. The method was tested through the 400 ORL human face database. The method achieved sensitivity of 84.6%, specificity of 93.4% and accuracy of 88.45%.

With image based passive methods some researchers like Nixon [27] proposed an approach for accurate measurement of eye spacing using Hough transform due to the circular shape of iris.

Similar to the eye, the iris detection has from and multi structure complexities as noted below:

1. The size and form of iris is variable
2. The effect of eyelids and eyelashes is complicated
3. There are deformations due to changes in pupil's size

In an effort to justify our approach, the iris detection algorithms proposed by John Daugman [5], Richard Wildes [15], and Ya ping [24] are noted in the following passages.

An integro-differential operator to isolate the iris/sclera and iris/pupil boundaries was first utilized by Daugman [5]. The same integro-differential with arcuate contours (arcs) was applied to localize eyelids.

However, in cases where there were noises in the eye image, such as from reflections, this integrodifferential operator seems to be failing.

The algorithm of Richard Wildes [15] proposes another approach which grabbed high interest in the field. The method was divided into two steps; converting the image into a binary edge map based on the gradient-based edge detector, and applying the HT2 to detect the boundaries.

Another approach rather close to that of Daugman depended on the same integro-differential operator [24]. However, it made use of the canny operator to find the approximate boundaries first. Table 1 shows the summarized results of these approaches.

Table 1. Compare of selected iris detection methods

System	FPR	FNR
Daugman	0.09	0.01
Tisse	1.84	8.79
Li Ma	0.02	1.98

$$\text{Where } FPR = \frac{FP}{FP+TN} \quad (1)$$

$$\text{and } FNR = \frac{FN}{TP+FN} \quad (2)$$

- FPR=False Positive Rate
- FNR=False Negative Rate
- FP=False Positive= Incorrectly identified
- FN=False Negative= In correctly reject
- TP=True Positive= Correctly identified
- TN=True Negative= Correctly reject

Non-intrusive techniques, which were utilized in this work, capture images of the eyes or faces through cameras, utilizing active infrared-based or Image-based passive approaches. The active infrared-based approach uses the physiological properties of eye/pupils under infrared. [19]

wholesome of the image-based passive methods localize and track the eyes using the visual appearance features of the human eye. These approaches detect eyes based on the unique intensity distribution or shape of the eyes.

In [20] a method for tracking iris use Kalman filter is introduced. This method at first recognizes iris with genetic algorithm and then tracks its movements through Kalman filter.

3. THEORETICAL BACKGROUND

3.1. Histogram

A histogram is a way to graphically represent the distribution of data in a data set. Each data point is placed into a bin based on its value.

In gray-scale image, histogram is defined as a number of points in image per each level of brightness. In the other word, histogram is the cumulative distribution function (CDF) of image's brightness level.

3.2. Support Vector Machine

3.2.1 Linear SVM

Assume that there are L training points, where each input x_i has D dimension and is in one of two classes; $y_i = -1$ or $+1$, i.e. The training data is of the form:

$$\{X_i, y_i\} \text{ Where } i=1 \dots L \quad y_i \in \{-1, 1\}, X \in R^D$$

Here if the data is linearly separable, meaning that a line on a graph of x_1 vs x_2 can be drawn separating the two classes when $D = 2$ and a hyper-plane on graphs of x_1, x_2, \dots, x_D for when $D > 2$. [9,14] This hyper-plane can be described by $W \cdot X + b = 0$ where w is normal to the hyper-plane and the perpendicular distance from the hyper-plane to the origin is $\frac{b}{w}$

Support Vectors are the examples closest to the separating hyper-plane and the aim of Support Vector Machines (SVM) is to orientate this hyper-plane in such a way as to be as far as possible from the closest members of both classes. [14]

$$w = \sum_{1 \leq i \leq L} \alpha_i y_i x_i \quad (3)$$

$$f(x) = \sum_{1 \leq i \leq L} \alpha_i y_i (x_i, x) + b \quad (4)$$

3.2.2 Non-linear SVM

Non-linear support vector classifiers map input space X into a feature space F via a usually non-linear map $\phi: X \rightarrow F, x \rightarrow \phi(x)$ and solve the linear separation problem in the feature space by finding weights α_i of the dual expression of the separating hyper planes vector w :

$$w = \sum_{1 \leq i \leq l} \alpha_i y_i \phi(x_i) \quad (5)$$

$$f(x) = \sum_{1 \leq i \leq l} \alpha_i y_i (\phi(x_i) \cdot \phi(x)) + b \quad (6)$$

Usually F is a high-dimensional space where images of training samples are highly separable, but working directly in such a space would be computationally expensive. [26]

The decision function can be computed with kernel function $K(x, y)$ and it can also be shown that finding the maximum margin separating hyper plane is equivalent to solving the following optimization problem. [13]

$$\text{sgn}(f(x)) = \text{sgn}[\sum_{1 \leq i \leq l} \alpha_i y_i K(x_i, x) + b] \quad (7)$$

Where the bias b is computed as:

$$\sum_{1 \leq i \leq l} \alpha_i y_i = 0 \quad (8)$$

3.3. Hough Transform

The Hough transform is a technique which is used to isolate features of a shape in an image. The commonly usage of HT is detection of regular curves such as line, circles, and ellipses. For the first time patent of Paul Hough in 1962 [4] released Hough Transform theory and after that in 1972, Richard Duda and Peter Hart [6] introduced HT as a today's applications.

$$Y = mx + b \quad (9)$$

$$b = -mx + y \quad (10)$$

The procedure first finds the intensity image gradient at all the locations in the given image by convolving with the sobel filters. The gradient images along x and y direction, is obtained by kernels that detect horizontal and vertical changes in the image. The sobel filter kernels are:

$$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (11)$$

$$G_y = \begin{bmatrix} -1 & 0 & 1 \\ 2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (12)$$

The absolute value of the gradient images along the vertical and horizontal direction is obtained to form an absolute gradient image using the equation.

$$G_{abs} = G_x + G_y \quad (13)$$

$$G_{abs} = \sqrt{G_x^2 + G_y^2} \quad (14)$$

The absolute gradient image is used to find edges using Canny [25]. The edge image is scanned for pixel (P) having true value and the center is determined with the help of the following equations:

$$x_c = X - r \cdot \cos(\theta) \quad (15)$$

$$y_c = Y - r \cdot \sin(\theta) \quad (16)$$

Where X, Y are the coordinates at pixel P and r is the possible range of radius values, θ ranges from $[0: 2\pi]$

3.4. Kalman Filter

The Kalman filter has been used extensively in Computer Vision research. Most of the applications that utilize Kalman filter focus on the problems of object tracking and structure from motion [28]. The general approach of Kalman filter is as follow:

For $k = 0$

- Initialization = x_0, P_0

For $k = 1, 2, \dots$

- Predictor equation

$$x_{k+1} = Ax_k + Bu_k + w_k$$

- Calculation of prediction minimum mean square error matrix

$$P_{k+1} = A P_k A^T + R_k$$

- Computing vector Kalman gain

$$K_k = A P_k C^T (C P_k C^T + S_z)^{-1}$$

- Correction equation

$$\hat{x}_{k+1} = (A \hat{x}_k + B u_k) + K_k (y_{k+1} - C \hat{x}_k)$$

$$P_{k+1} = A P_k A^T + S_w - A P_k C^T S_z^{-1} C P_k A^T$$

END

4. METHOD

One of the goals of this work is to realize automatic detection and tracking of irises in a video sequence taken in a normal indoor environment without using any special lighting. This work consists of five stages; video acquisition, face, eye, and iris detection and iris tracking using Kalman. Finally, as a final step, a decision may be made to distinguish normal from abnormal behaviour in regard to tracked iris movements. Such decision is made by means of comparing the gaze direction to that of the target movement.

4.1. Video Acquisition

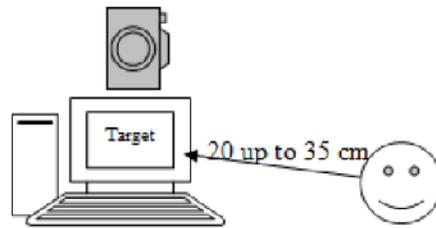


Figure1. Video acquisition schematic

Image acquisition is the first step, and a critical one, in any biometric identification system. Obtain a perfect noise-free video is highly desirable to minimize any chances of misidentification. Video is based on using an HD camera, with its resolution set to 680x420, and the type of the video set to Avi. Videos are taken under natural light.

To be able to process the image, we should first get its RGB representation then change to gray level representation.

According to the above figure, tested person sit in front of monitor with the distance of 20 up to 35 cm.

4.2 Face Detection

In recent years face recognition is known as an attractive issue in many potential applications in machine vision, communication, and control system. So many various and robust methods are introduced for achieve this approach.

As mentioned in section 2, some approaches have been proposed to combine the VJ face detector [7] with skin-color detector, which are mainly based on pre-filtering. [9]

This paper presents the hybrid method for face detection in color frames.

In this work detection is divided in two; pre-processing and detecting.

In the preprocessing the skin color detection has been applied. It converts RGB frames to L.a.b color space for segmenting image into two parts; skin color and non-skin color space with threshold of 1.8.

The converting RGB to L.a.b space is done in two phases:

1. RGB to XYZ

$$[X] = [0.412453 \ 0.357580 \ 0.180423] * [R] \quad (17)$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (18)$$

2. XYZ to L*a*c

$$L = \begin{cases} 116 * (\frac{Y}{Y_n})^{\frac{1}{3}} - 16 & \text{for } Y/Y_n > 0.008856 \\ 903.3 * \frac{Y}{Y_n} & \text{otherwise} \end{cases} \quad (19)$$

$$a^* = 500 * (f(X/X_n) - f(Y/Y_n)) \quad (20)$$

$$b^* = 200 * (f(Y/Y_n) - f(Z/Z_n)) \quad (21)$$

$$\text{Where } f(t) = \begin{cases} t^{1/3} & \text{for } t > 0.008856 \\ 7.787 * t + 16/116 & \text{otherwise} \end{cases}$$

Here X_n , Y_n and Z_n are the tristimulus values of the reference white.

The reverse transformation (for $Y/Y_n > 0.008856$) is

$$X = X_n * (P + a^* / 500)^3 \quad (22)$$

$$Y = Y_n * P^3 \quad (23)$$

$$Z = Z_n * (P - \frac{b^*}{200})^3 \quad (24)$$

$$\text{Where } P = (L^* + 16) / 116$$

At the second part of scenario, VJ method selects the true face region among candidates. Haar-like features and skin color detection are combined to achieving high performance. [9]

The VJ method uses Haar-like features and AdaBoost learning algorithm. [7] Sequential of weak classifiers have been used in the approach introduced with them. The use of a cascade of lassifiers made this approach one of the first real-time frontal-view face detection methods. [16]

Indeed, color is a low-level cue that can be implemented in a computationally fast and effective way for locating objects. [16] It also offers robustness against geometrical changes under a stable and uniform illumination field. In some cases, it can clearly discriminate objects from a background.

In figure some results which obtained in this work are shown:

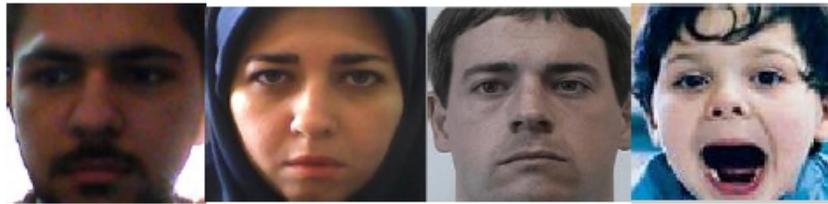


Fig.2. Results of hybrid face detection

After that face frames convert to gray-scale and resize to 200x200 pixels and prepare for next stage.

4.3 Eye Detection

In recent years eye detection has attracted much attention and its research has rapidly expanded by not only engineers but also neuroscientists, since it has many potential applications in computer vision communication and automatic access control system.

Since, eyes have a dynamic and inconstant model, detecting them in face image isn't easy job and modelling eye in unit statics is almost impossible. So in most cases, researchers use more than one mechanism for eye detection.

This part describes an attempt to build a component based eye detector using support vector machine classifiers.

We take a straightforward approach in implementing SVM₃ classifier that detects eyes in gray scale face images according to histogram features of them. This method can find eye region in 200x200 frame size with accuracy of 97.6% in 0.38 second.

Moreover we improve the accuracy and process time with face segmentation in three parts, and only one part of face is given to SVM classifier as face frame.

After face resizing in all frames, with using of face geometric, the region around eyes can be isolated. This operate causes search region 4.45 times smaller, and the speed is raising in 1.73 times larger.

Eye detection based only face anatomy or histogram may fail in different conditions, three failed results are shown as follow:

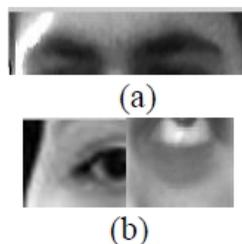


Fig.3. Failed results in histogram based method (a) and anatomy based method (b)

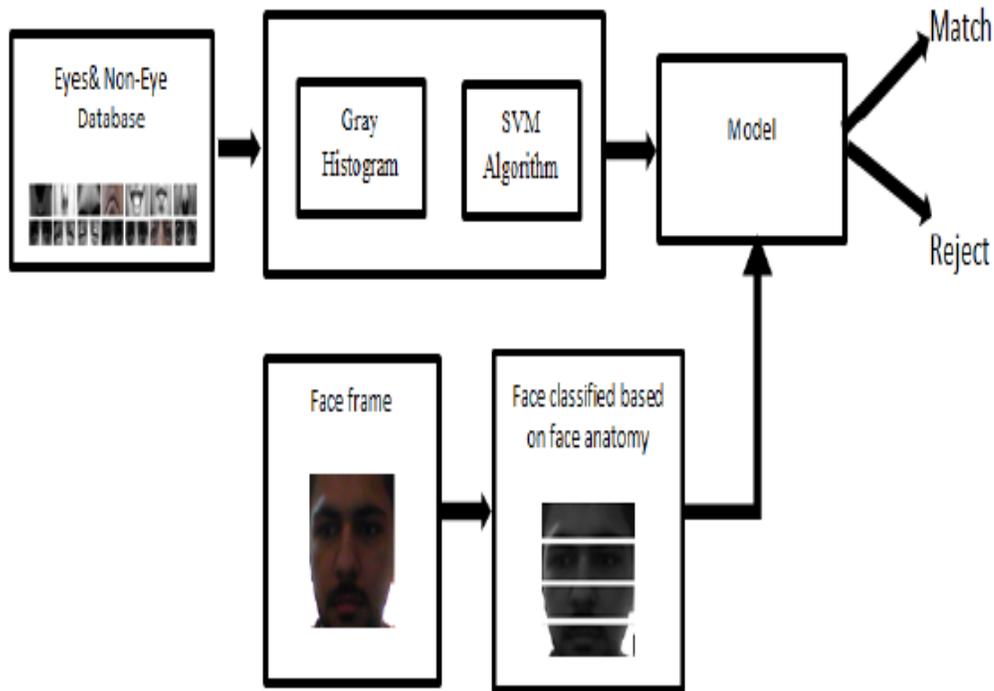


Figure.4. Our eye detection method

Above figure illustrates the stages of proposed eye detection. Our data base is consists of eye and non-eye images which are product by us. Eyes database were marked in Cohn-Kanade (CK) face image data base, [10] using Viola-Jones [7] eye detection method in 100 training images of human face and extracted to 100 positive samples, and was resized to 40*20 pixels each, represented in gray scale images. Negative samples for training were generated randomly from a set of face images without areas that contain positive samples (eyes).



Fig.5. Some templates used for training SVM

Obtained histogram of databases are given to system to extract features based on peak detection, [12] and the number of peaks represents the number of distinct regions. [22] The stages of operating peak detection due to histogram of gray-level image are as follow:

- For each gray level i in image, put the element (i, f_i) into the set H_0 , where f_i is the frequency corresponding to the i th gray level.
- Compute the set H_1 , which contains the elements representing the points of local maxima in the image histogram.

$$H_1 = \{(i, f_i) | ((f_{i-1} < f_i) \& (f_i > f_{i+1}))\} \quad (25)$$

Where $(i, f_i) \in H_0$

- Compute the set H2 comprising of elements having frequency higher than 1% of the maximum frequency f_{max} in the set H1.

$$H2 = \{(i, f_i) \mid f_i > f_{max} \} \quad (26)$$

Where $(i, f_i) \in H1$

- Remove the elements having close peaks from set H2. This is done by checking the difference between the gray levels of two elements in set H2.

$$H3 = H2 - \{((i, f_i) \mid i > j, (i - j) \leq 20) \& (f_i = \min(f_i, f_j))\} \quad (27)$$

- The number of elements in set H3, denoted as $|H3|$, is equal to the number of significant peaks in the histogram:

$$n = |H3|$$

Where n is the number of prominent peaks. [12]

After that SVM is learned according to distinct region with label 1 and otherwise other regions give label with value 2. At last, as mentioned, for faster operation instead face frame, more than quarter of face region give to model to choose real eye region.

4.4 Iris detection, localization, and tracking

In this section, we describe iris detection from the extracted right and left eye regions. Such as the previous sections, at first each of eye regions must be resized in same size. This operation leads to correct comparison between eye displacements.

The strategy of iris recognition is as follow:

- pre processing
- Iris detecting in extracted eyes
- Tracking iris movement

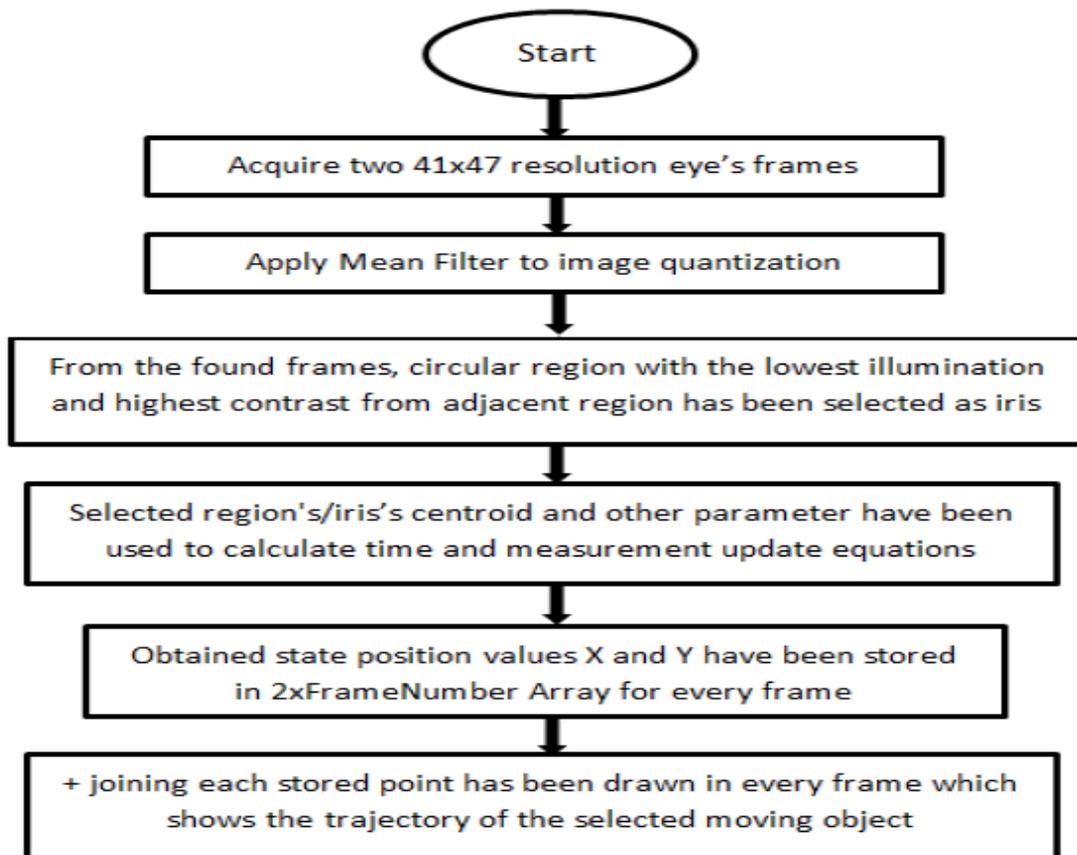


Fig.6.Flowchart of Iris detection

4.4.1 Pre-processing

Light reflections in the eye images negatively effect on iris border detection. Several image enhancement techniques have been used, such as contrast enhancement and histogram equalization, to enhance the input image before applying the detector.

We do pre-processing in order to eliminate noise and enhance contrast of image to more accurate identification.

External noise is removed by blurring the intensity image. So in this paper, we apply mean filter to denoising. Mean filter operates in two moods; Averaging and as Convoluting.

We're careful to not lose the original image, so we apply 3x3 filter in convoluting mood of mean filter:

$$g(i,j) = \frac{1}{n.n} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}_{n \times n} \quad (28)$$

Where $n=3$

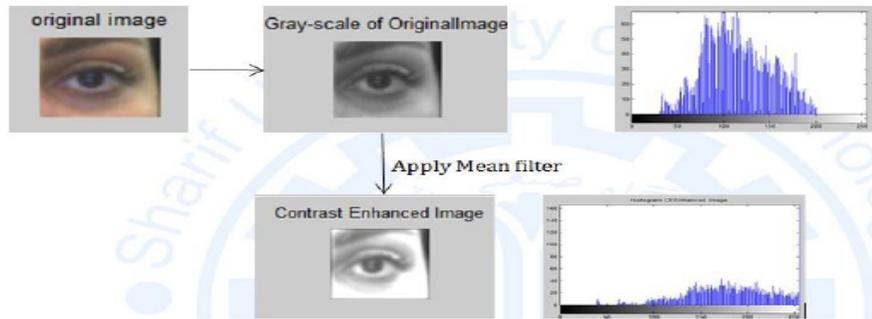


Fig.7. apply mean filter as pre-processing

4.4.2 Iris detection in extracted eyes

As pupil and iris are the largest circular areas with lowest illumination, circular Hough Transformation can be used for iris's candidate detection. According to our experiment the radius of iris in 41x47 pixels eye frame, is between 5 to 10 pixels, so in order to reduce iris's candidates to improve the speed and accuracy of process, the radius is limited in a defined range.

The Hough transformation is able to overcome shadows and noise. The approach is found to be good particularly dealing with all sorts of difficulties including severe occlusions [27].

As mentioned in section 3, a maximum point of Hough space will correspond to the centre coordinates of the circle. The center of the concentric circle can be located at the peak of the summation of initial center points over the radii of interest:

$$X_0 = \sum_{rmin}^{rmax} a \quad (29)$$

$$Y_0 = \sum_{rmin}^{rmax} b \quad (30)$$

Thus, the center of iris is obtained as follow:

$$iris \ coordinate = (max(X_0), max(Y_0)) \quad (31)$$

Iris coordinates are saved to introduce to Kalman filter in tracking step.

4.5. Eye/Iris Tracking

Eye movements' tracking is a research methodology that is used to examine visual attention and other cognitive processes in a variety of areas, including scene perception, visual search. The use of eyetracking data in psychologically oriented research rests on the assumption that overt attention (as manifested by the exact eye location) and covert attention are tightly linked, [29] in other words, that there is a close relationship between the eyes and the mind. [22]

In this paper the eye tracker applies which gives some information about normality and abnormality of brain reaction with regard to two basic components of eye movement behaviour:

eye fixations (i.e. where and for how long a subject is looking) and saccades or eye movements (i.e. where the eyes move next).

To achieve this purpose, we apply Kalman filter as a tracker to pursuit the eye movement tragedy in a recorded video.

We characterize the pursuit eye movement by the following state-space equations for right and left eye respectively:

$$\begin{bmatrix} \dot{X}_{1R} \\ \dot{X}_{2R} \\ \dot{V}_R \end{bmatrix} = \begin{bmatrix} 1 & dt & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{1R} \\ X_{2R} \\ V_R \end{bmatrix}$$

$$\begin{bmatrix} \dot{X}_{1L} \\ \dot{X}_{2L} \\ \dot{V}_L \end{bmatrix} = \begin{bmatrix} 1 & dt & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{1L} \\ X_{2L} \\ V_L \end{bmatrix} \quad (32)$$

$$Z_R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{1R} \\ V_R \\ X_{2R} \end{bmatrix}$$

$$Z_L = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{1L} \\ V_L \\ X_{2L} \end{bmatrix} \quad (33)$$

According to the theory of Kalman filter, X is the state vector at the next frame $t+1$, linearly relates to current state X .

In this research, X_{1R} , X_{2R} , and V_R are coordinate and velocity of right eye, and also X_{1L} , X_{2L} , and V_L are coordinate and velocity of left iris respectively. dt is the interval between two frames. In some methods such as [11] only the center coordinate of left iris is assumed and another one is determined according to distance of two irises. But in our research due to the high sensitivity of detection and inconsistencies in how moving the eyes of some autistic children, both of the right and left iris's centre coordinates are calculated. And give to Kalman filter as observations.

At last, according to the iris displacement in x axis than first frame, the gaze and eye movement direction is obtained.

Due to the similarity of the optical behaviour of tested person according to moving target in video playback, we realize the brain's ability to communicate with her/his environment and eyes.

Obtained results in next section show the accuracy of our diagnose theory.

5. Results and conclusion

Robust and non-intrusive human eye and iris tracking problem has been a fundamental and challenging problem for computer vision area. Designing a Robust and Non-intrusive eye

tracking based on eye movement and gaze direction is the main goal of this paper. The results can be used to diagnose some abnormality such as Autism Spectrum Disorder.

This aim is achieved by pursuit eye movement in faced with moving and stationary targets.

To achieve this purpose we have to do several steps to obtain information about iris/eye movement. Table 3 illustrates the accuracy and speed of each of the stages:

Table.3.The final results in time and accuracy

Methods	Frame size	Processing time (frame num/sec)	Processing time in 207 frames (Sec)	Accuracy
Face detection	640x480	1.65 (60.5)	138	100%
Eye Region detection	200x200	4.5	45.54	99.1%
Right and Left Eye Detection	191x41	11	18.8	100%
Iris detection in each eyes	47x41	13.157	15.732	96.42%
Iris/Eye Tracking	47x41	14.79	13.98	97.8%

According to the process times, iris detection method can be able to detect 13 frames per one second (fps).

In this part we show results of our approach step by step. The accuracy in all stages is calculated due to the number of acceptable results.

$$Accuracy = \frac{TP+TN}{total\ number\ of\ frames} \quad (34)$$

At first, face detection based on skin color and adaboost classifier, in fix state of head, has significant role in reducing processing time and increase accuracy. Table shows the compare of three methods of face detection:

Table.4. Compare of two face detection method and proposed method

Face Detection Method	Based on skin color	Viola and Jones	Proposed method
Frame size	640x480	640x480	640x480
Accuracy in head straight state	96.9%	100%	100%
Frame num/sec (fps)	1.68	1.45	1.65
Total time in 207 frames (sec)	132	142	138
Total time in 380 frames (sec)	141.07	152.08	137.89

In table 4 we can see that the color based system is simpler, faster but this simply in computational can be caused some mistakes in occlusions and sometimes it is sensitive to illumination changes and background environments. In contrast, Haar-like based system is slower, but more effective. [7] So Working on segmented region with skin color would be easier, faster and accordingly system would have higher correct detection rate.

In the next stage of our research, right and left eyes are detected in 200x200 gray-scale face frames. We mix two methods, based on anatomy and based on histogram features of eye region. In Below table, results which are achieved by each of the methods in our database are illustrated.

Table.5. Compare of two face detection method and proposed method

Eye Detection Method	Based on Face Anatomy	Histogram feature based	Proposed Method
Frame size (Pixel)	200x200	200x200	200x200
Accuracy in direct pose of head	98.72%	97.6%	99.1%
Frame's num/Time (fps)	5.16	2.6	4.5
Total time in 207 frames (Sec)	40.08	78.3	45.54

In our methodology in compare with histogram based method, the region search becomes 5.16 times smaller, so the speed increased.

The some samples of our successful results are shown below:

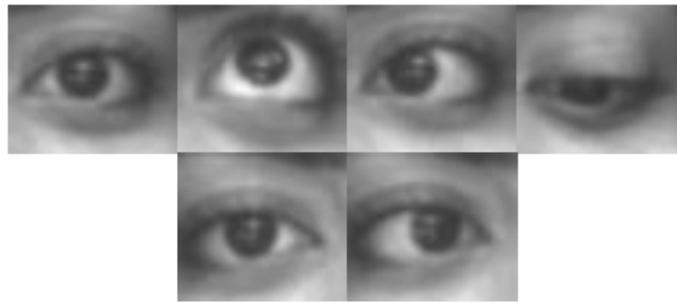


Fig.8. Some successful results of right and left eye detection

Note that in this research the accuracy of results is much important than high speed in doing process.

To achieve suitable tracking, one of the important stages is obtain accurate iris detection as initial condition of Kalman. So before applying iris detector in eyes frames mean filter is implemented to enhance contrast in eye frames. The pre-process is useful to increase the precision of recognize. In table 6, one result before and after applying mean filter on eye frame is shown.

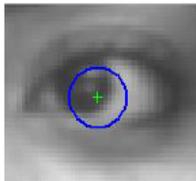
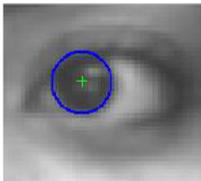
Frame#90 (RightEye)	Real Coordinate	Frame#90
		
(a)	30, 24	(b)
37.329561, 24.419539		28.877506, 23.990491

Table.6. Efficacy of using mean filter in eye frame

Table shows the total results of detection methods in time and accurate:

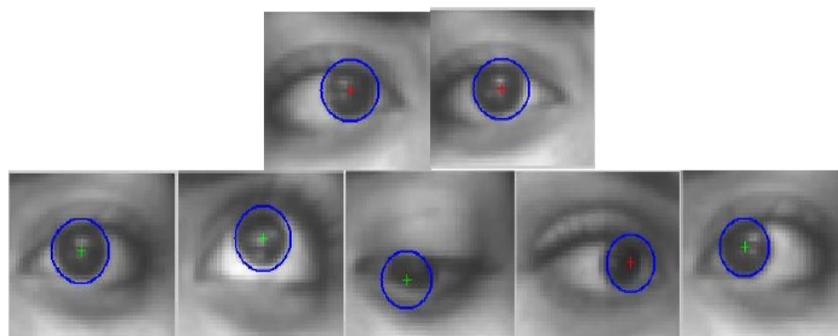


Fig.9. Examples of successful images

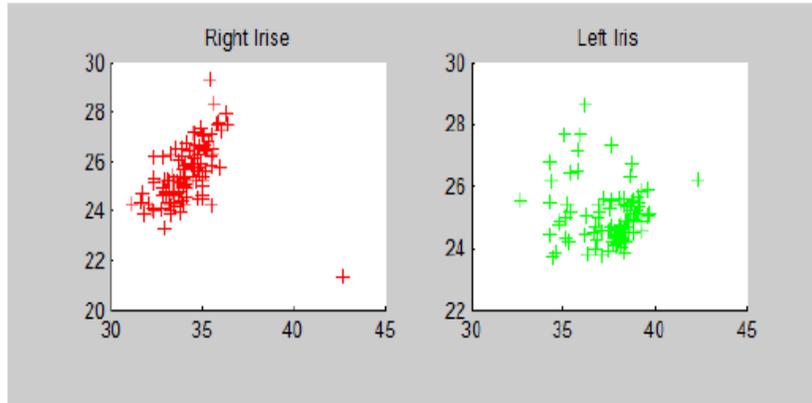


Figure.10.Iris coordinates' Spatial Distribution in 41x47 eye frames

Then the saved coordinates are given to kalman .Kalman according to initial state and estimation finds the coordinates of iris in each frame and track the eye movement strategy. So, the output of tracker includes of information about saccades, fixation, and velocity of eye/iris during watching movie.

To test the performance of the proposed detector and tracker, we have compared the correct position of the eyes with obtained results of detection and tracking steps.

Table.7. Tracking results

Frame Num.	True values			Estimates			Error in Estimate		
	Position	Position of Left Iris	Gaze Direction	Position of Right Iris	Position of Left Iris	Gaze Direction	Position of Right Iris	Position of Left Iris	Gaze Direction
t=K1	$X_1(k)$ $Y_1(k)$	$X_l(k)$ $Y_l(k)$	G_D	$\hat{X}_1(k)$ $\hat{Y}_1(k)$	$\hat{X}_2(k)$ $\hat{Y}_2(k)$	\hat{G}_D	$E_{RX}(k)$ $E_{RY}(k)$	$E_{LX}(k)$ $E_{LY}(k)$	
7	20.2316 19.9449	23.8493 19.8346	Looking Frontal	20.122280 20.160475	24.639898 20.802021	Looking Frontal	0.1 0.21	0.79 1.03	True
36	19 21	21.4375 19.0404	Looking Right	19.254528 20.762911	23.780096 20.888066	Looking Frontal	0.25 0.238	2.34 1.84	False
50	25.3 21	28 19	Looking Left	25.869202, 20.775230	28.186941, 20.350450	Looking Left	0.569 0.224	0.186 1.350	True
82	21.414437 20.661996	22.105824 19.196971	Looking Frontal	21.1 21	22 18.2	Looking Frontal	0.31 0.34	0.105 0.996	True
99	25 22	26.5 19	Looking Left	24.786070, 20.747585	25.1 20.342	Looking Left	0.214 1.253	1.4 1.342	True

In this research, the iris displacement is obtained by differential between each frame and the first frame (iris coordinate in direct pose). In this case five states of gaze are occurred:

- Looking Frontal for $-1.6 \leq x \leq 1.6$ & $-1.6 \leq y \leq 1.6$
- Looking Right for pixel $x > 1.6$

- Looking Left for $x < -1.6$ pixel
- Looking up for $y > 1.6$
- Looking down $y < -1.6$

Which in table 7 gaze direction is calculated only due to x .

The results are shown some errors caused by detection are corrected in this stage. In order to avoid this problem, the gaze direction will be checked every two frames, to remove the caused error in one frame.

Now in iris detection and tracking method we achieved about 97.8% accuracy in iris detection as observation of Kalman filter. The error function is defined as:

$$\text{Error} = \begin{cases} 1 & \text{if Distance} > 1 \\ 0 & \text{if Distance} \leq 1 \end{cases}$$

Where Distance is Euclidean distance that can be used to calculate distance between the center of circles and approximate centre.

$$\text{Distance} = |\sqrt{(X_{RR} - X_{OR})^2 + (Y_{RR} - Y_{OR})^2}| \quad (35)$$

$$\text{Distance} = |\sqrt{(X_{RL} - X_{OL})^2 + (Y_{RL} - Y_{OL})^2}| \quad (36)$$

Where X_{RR} , Y_{RR} , X_{RL} , and Y_{RL} are coordinates of left and right iris center and X_{OR} , Y_{OR} , X_{OL} , and Y_{OL} are coordinates of approximate left and right iris centre.

On the other hand it should be noted that in all normal cases, both eyes are confirming and verifying each other in order to move in the direction of gaze. Knowing this fact can be a way of checking accuracy and properly of detection and tracking irises of normal person. These two figures and are shown the displacement of right and left irises in X and Y axis respectively. As expected, red and green points verify each other.

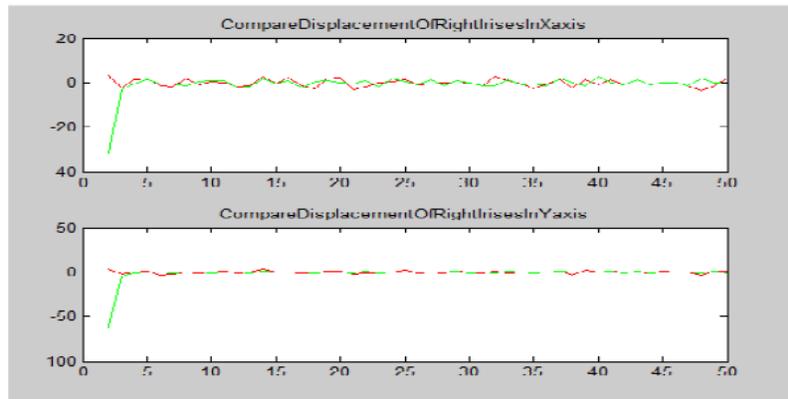


Fig.11. right (Red) and left (Green) irises displacement in X and Y-axis

According to the figure 11, in normal person, two eyes move same as each other; in direction and displacement.

Finally, according to the gaze tracking, the displacement of iris relative to the direct pose of gaze is obtained (figure 12). In this test, target in video, moves to the left and right side in different frames with different speed (green diagram). Unfortunately, autistic is unable to pursuit target in some frames and in high speed (red diagram). The speed of iris is calculated due to this formula:

$$velocity = \frac{\Delta x}{\Delta f} \quad (37)$$

Where Δx is displacement of iris in x axis in Δf frames.

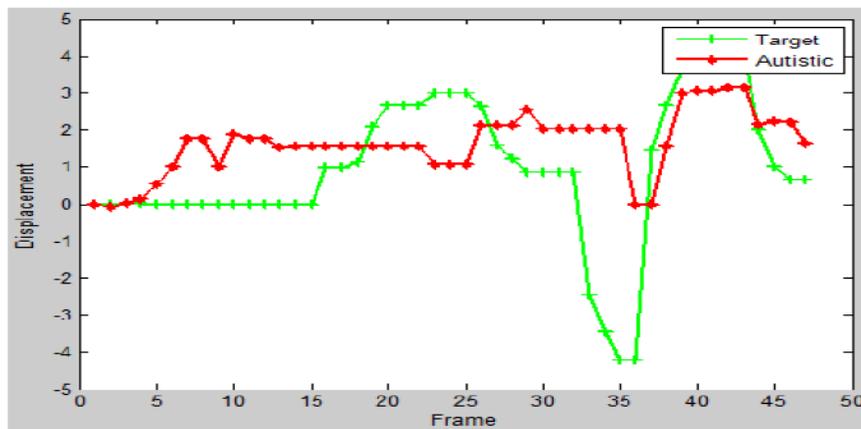


Fig.12 target and Autistic's iris displacement in x-axis at the same time in the first fifty frames

REFERENCES

- [1] Catherine Saint-Georges, AmmarMahdhaoui, Mohamed Chetouani, Raquel S. Cassel, Marie-Christine Laznik, Fabio Apicella, PietroMurator, Sandra Maestro,FilippoMurator, and David Cohen, Do Parents Recognize Autistic Deviant Behavior Long before Diagnosis? Taking into Account Interaction Using Computational Methods, PLoS One. 2011; 6(7): e22393.
- [2] American Psychiatric Association. Autism spectrum disorder, In: Diagnostic and Statistical Manual of Mental Disorders, Fifth Edition, American Psychiatric Association, and Arlington, VA 2013, p.50.
- [3] Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, ISBN-10: 0201180758 | ISBN-13: 978-0201180756 | Publication Date: January 15, 2002 | Edition: 2nd, 2002.
- [4] Hough P V C, U. S. Patent, Method and means for recognizing complex patterns, 3069654, 1962.
- [5] J. Daugman. How iris recognition works. Proceedings of 2002 International Conference on ImageProcessing, Vol. 1, 2002.
- [6] Duda, R. O. and P. E. Hart, "Use of the Hough Transformation to Detect Lines and Curves in Pictures," Comm. ACM, Vol. 15, pp. 11–15, 1972.
- [7] P. Viola and M. Jones, Rapid object detection using a boosted cascade of simple features, in Computer Vision and Pattern Recognition, pp. 511–518, 2001.