VOL. 11, NO. 15, AUGUST 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

# ADVANCED MOTION TRACKING BASED MOBILITY ASSISTANCE FOR PHYSICALLY DISABLED

Satyanarayana P., Sai Prajwal K., Chandra Naga Varma T., Sri Manojna E. and Sitara S.
Department of Electronics and Communications Engineering Koneru Lakshmaiah University Green Fields, Vaddeswaram
E-Mail: satece@kluniversity.in

## ABSTRACT

In present-day scenario, many people aren't able to control powered wheel chair using various interfaces like joystick, head control or voice control. For this reason, a new image-processing based mobility assistance system is proposed in this paper that automatically tracks the leg movement of the assisting person and follows him accordingly. In addition to this, an eye tracking feature has been inculcated into the proposed system that would help the disabled person in controlling the movement of wheel-chair using eye movement in the absence of an assisting person. The leg tracking module has been devised using the background subtraction and CamShift algorithms while the eye-tracking module utilizes the Haar cascades along with the Daugman's algorithm to track the eye-movement.

**Keywords:** motion tracking, background subtraction, CamShift algorithm, haar cascades, daugman's iris localization algorihm.

## 1. INTRODUCTION

Census 2001 has uncovered that more than 21 million individuals in India are suffering from one or the other sort of disability. This equals to 2.1% of the population [1].

| **Category** | **Population** | **Percentage (%)** |
|---|---|---|
| Total population | 1,028,610,328 | 100 |
| Total disabled population | 21,906,769 | 2.1 |
| **Type of disability** | | |
| In seeing | 10,634,881 | 1 |
| In speech | 1,640,868 | 0.2 |
| In hearing | 1,261,722 | 0.1 |
| In movement | 6,105,477 | 0.6 |
| Mentally disabled | 2,263,821 | 0.2 |

**Figure-1.** Census 2001 data corresponding to disabled population in India.

This data helps us realize the urge in the requirement of Assistive technology for the disabled population. In recent years, there has been a sharp increase in the Assistive technology development for the people with the above mentioned disabilities, and great progress has been made in communication systems between humans and computers [2].

These advances have been made mostly in communication from machines to humans by the methods of GUI or multi-media applications. These advances in correspondence have been unobtrusive, utilizing consoles, mouse, joysticks and touch screens. The current computer input devices can be isolated into five classes:

a) Bio-potential based strategy which uses potential from user's body actions which are acquired by utilizing special instruments such as Electrooculography [3], Electromyography (EMG), Electroencephalograph (EEG) [4] and Search coil. While the EOG method [5] uses the voltage differences between the fore and aft surfaces of the eye, the pursuit coil output can be utilized as a source of PC data for the differently abled individual.
b) Voice Based strategy [6] using the user's voice as an input. Voice analysis is performed to dissect user's voice and convert into digital data. The shortcoming of this framework is that it is vulnerable to noise. Different voices which originate from encompassing users might influence the framework.
c) Motion based technique [7], uses other typical movement organs to provide the input. For example, Head movements in some cases are used as an input to the assistive system.
d) Image Analysis technique [8]-[13], uses a camera to dissect user's desire and converts accordingly into computerized information. There are quite a lot of methods that help in analyzing the user's desire. One among them is the Gaze-based [14] movement assistance.
e) Search coil strategy [15] using induced voltage with coil by appending the contact lenses to user's eyes.

As incidentally weakened people like spinal cord hurt subjects and developmentally disabled subjects with motor loss of movement, for instance, amyotrophic even sclerosis (ALS) and Guillain–Barre Syndrome [16] face issues in conveying their desires because of the affected motor neurons that control the voluntary muscles, only some of the above mentioned methods will be of their help. To serve the people who fall under this category, this paper suggests a suitable system
The proposed system has two modules:

a) Leg-tracking module
b) Eye-tracking module

Leg tracking module involves tracking the legs of the assisting person walking ahead of the mobility assistive system. This module serves 0.6% of population who has disability in movement and can also support

ARPN Journal of Engineering and Applied Sciences



motion for 1% of population [1] who have the disability for seeing as it is hard for them to move by themselves.

Eye tracking includes tracking of the eye ball movement and mobility support accordingly. This module is very supportive to the disabled in absence of an assistive person provided that they have proper eye movement. There are significant benefits of the proposed system when compared to the state-of-art systems. The bio potential based strategy and motion based technique involves uses body and organs actions whereas voice strategy needs voice input to the system which are strenuous as it is burden on the users to do so. Hence the proposed system deploys image analysis technique which employs a camera for tracking and consequently analysing them using different image processing algorithms.

Leg tracking module involves two algorithms for tracking the leg movement of assistive person i.e., Background Subtraction and CamShift techniques while eye tracking module employs two image processing algorithms haar cascades and Daugman's iris localisation for tracking the eye movement.

## 2. KEY CONCEPTS

### 2.1. Background subtraction algorithm

To identify the moving objects we use a technique called Background subtraction. It is a crucial technique in many motion detection applications and also a major pre-processing step in many vision based applications. Technically we use this process to extract the moving foreground from background. This method calculates the foreground mask by subtracting the previously captured frame from the current frame.

Let $I(\mu, \vartheta, \tau)$ be the current frame taken at an instant $\tau$ and $\mu, \vartheta$ be the pixel co-ordinates of the image. Here,

$\mu \in [0, M-1]$, M being the height of the image.
$\vartheta \in [0, N-1]$, N being the width of the image.

$\xi$ is the time difference between two consecutive framing instances. $\xi$ is dependent on $f$ (Frames Per Second) parameter settings of the Capturing device.

$$\xi = 1/f$$

Now the resultant image $J(\mu, \vartheta, \tau)$, obtained by subtracting the previous frame with the current frame is,

$$J(\mu, \vartheta, \tau) = I(\mu, \vartheta, \tau) - I(\mu, \vartheta, \tau - \xi)$$

Now the resultant image $J(\mu, \vartheta)$ is subjected to the process of thresholding so as to elevate the portions of the image holding the movement information. Portions of the image which doesn't have significant motion will be holding a pixel value **0**. The portions of the resultant image having the movement information will certainly have a non-zero pixel value. Thus, we set all those pixels holding significant non-zero values to 255.

The resultant image obtained by thresholding the mask image $J(\mu, \vartheta, \tau)$ is given by,

$$\varphi(\mu, \vartheta, \tau) = \begin{cases} 255; & J(\mu, \vartheta, \tau) \neq 0 \\ 0; & J(\mu, \vartheta, \tau) = 0 \end{cases}$$

$J(\mu, \vartheta, \tau)$ is pixel value of the resultant image that is obtained after subtracting the previous frame from the present frame.

$\varphi(\mu, \vartheta, \tau)$ is the manipulated pixel value obtained from its relation with $J(\mu, \vartheta, \tau)$. Some sample images (person moving in front of camera) of $\varphi(\mu, \vartheta, \tau)$, obtained after the background subtraction technique are provided in the following figure.

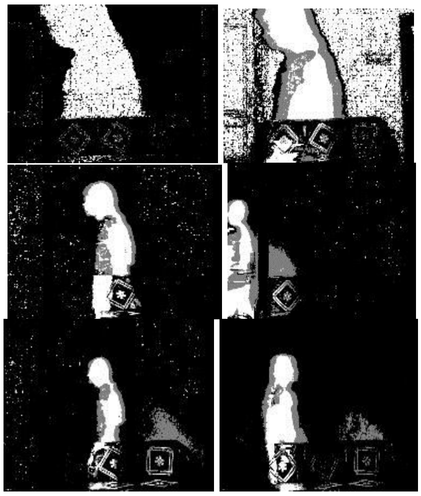

**Figure-2.** Resultant images of $\boldsymbol{\varphi(\mu, \vartheta, \tau)}$ obtained after background subtraction technique.

### 2.2. CamShift algorithm for object tracking

CamShift is a colour tracking algorithm which was initially developed by Gary R. Bradski [18] of Intel Corporation for tracking human faces. This algorithm is a step forward to Mean shift algorithm [22] and improvises on the limits of fixed size windows. This section 2.2 uses some key derivations provided by Gary R. Bradski [19] for utilizing CamShift Algorithm. Fast and robust approaches [20], [23] for CamShift tracking were later developed.

The generalized algorithm for CamShift [21[1] for Video Sequences can be given in the following way:

a) Choose the entire image as the region for which probability distribution is to be calculated.
b) Select the search window for MeanShift at required location.
c) For the region centred at the MeanShift search window, find the colour probability distribution.
d) Find the centroid of probability intensity of the current frame through MeanShift iterations and capture the zeroth moment (area) and mean location.
e) In the next frame, change the centre of window to centroid location found during the previous step and manipulate the window size as function of zeroth moment. The move to Step3.

To find the colour probability distribution (probability image) as mentioned in step3, we require a method which assigns the probability of occurrence of pixels in target zone to the corresponding pixel values. Histogram Back Projection is generally used for this purpose. Hue channel in HSV colour space is used by Bradski (1998) for histogram calculations. This minimizes the computation complexity for tracking. Moreover, we can also use histograms with multi dimensions having colour spaces such as RGB, CMY and HIS in cases where accuracy is the primary factor.

The input parameters for CamShift algorithm can be calculated by finding the first and second moments of the probability image. The corresponding equations are given as follows.

$$M_{00} = \sum_x \sum_y I(x,y)$$
$$M_{10} = \sum_x \sum_y xI(x,y)$$
$$M_{01} = \sum_x \sum_y yI(x,y)$$

Where $I(x,y)$ represents the probability of pixel at $(x,y)$ in the image and the range of $x$ and $y$ depends on the size of the image (frame). Now, from the moment values, the centroid of mean shift window can be obtained by following equations.

$$x_c = \frac{M_{10}}{M_{00}} \qquad y_c = \frac{M_{01}}{M_{00}}$$

The centroid coordinates form the basis for tracking of objects and are correspondingly calculated for every iteration until convergence from the moments. These centroid values along with calculated moments form a critical part in CamShift process for readjusting the window size and orientation of major axis. Additionally in CamShift, we need to measure other moments which are defined by

$$M_{20} = \sum_x \sum_y x^2 I(x,y)$$
$$M_{02} = \sum_x \sum_y y^2 I(x,y)$$
$$M_{11} = \sum_x \sum_y xyI(x,y)$$

The major axis orientation $\theta$ with respect to previous level is given by equation

$$a = \frac{M_{20}}{M_{00}} - x_c^2$$

$$b = 2(\frac{M_{11}}{M_{00}} - x_c y_c)$$

$$c = \frac{M_{02}}{M_{00}} - y_c^2$$

$$\theta = \frac{1}{2}\tan^{-1}(\frac{b}{a-c})$$

Where a, b and c are intermediate variables considered for calculating orientation $\theta$. The Eigen values (which form the length [l1] and width [l2] of newly formed distribution around the centroid) are provided with equations.

$$l1 = \sqrt{\frac{(a+c) + \sqrt{b^2 + (a-c)^2}}{2}}$$

$$l2 = \sqrt{\frac{(a+c) - \sqrt{b^2 + (a-c)^2}}{2}}$$

**2.3 Haar c; assification cascades for face and eye tracking**

An effective object detection method utilizes Haar feature-based cascade classifiers which was proposed by Paul Viola and Micheal Jones [17] using machine learning based approach. In this, a cascade function is trained from a lot of positive and negative images. This will be further used to detect the objects in other images.

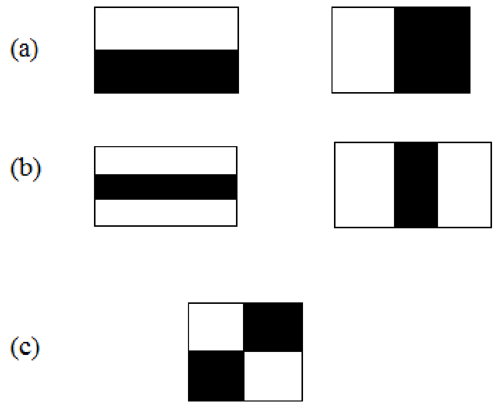


**Figure-3.** (a). Edge features, 3.(b). Line features and 3.(c). Four-rectangle features.

In case of face-detection, the algorithm requires a great deal of positive images (images with faces) and negative images (images without faces) to train the classifier. After this, we have to extract features from it. For this, haar features that are presented in the picture beneath are utilized. They are much the same as our convolutional kernel. Every feautre is a single value acquired by subtracting entirety of pixels under white rectangle from aggregate of pixels under black rectangle.

Now, all possible sizes and locations of each kernel are used to calculate a myriad number of features. Indeed, even a 24x24 window results more than 160000 features. To solve this, the integral images are introduced which simplifies calculation of the sum of pixels, how large may be the number of pixels, to an operation involving just four pixels.

$$ii(x, y) = \sum_{x' \le x, y' \le y} i(x', y')$$

Here, ii(x, y) is the integral image pixel at location (x, y). We see that this pixel value is sum of the pixels above and to the left of x,y, inclusive in the image **i**. Among all these features calculated, most of the features will be irrelevant. Selecting the best features among the 160000+ features is achieved by Adaboost.

For this, each and every feature is applied on all the training images. For each feature, the best threshold is found which will classify the faces to positive and negative. The features with a minimum error rate, corresponding to the features that best classify the face and non-face images, are selected. This results in a final setup having around 6000 features.

In order to still improvise it, the concept of Cascade of Classifiers was introduced. Instead of applying all the 6000 features on a window, the features are grouped into different stages of classifiers and are applied one-by-one. If a window fails the first stage, it is discarded. If it passes, the second stage of features are applied and the process is continued. The window passing all the stages is a face region. Authors' detector had 6000+ features with 38 stages with 1, 10, 25, 25 and 50 features in first five stages.

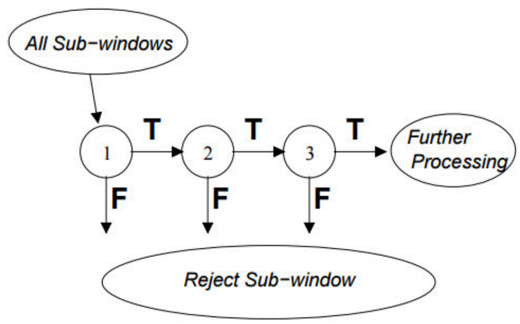


**Figure-4.** Schematic depection of a detection cascade.

### 2.4. Daugman's algorithm for finding the centre of pupil

An integrodifferential operator is proposed by Daugman to find both the pupil and the iris contour. In the first phase of the algorithm, Gaussian blur function is calculated. Gaussian blur mathematically is the convolution of image with a Gaussian function. The equation of a one dimensional Gaussian function is given by,

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{x^2}{2\sigma^2}}$$

Here, x is the distance from the point of origin in the horizontal axis while σ is the standard deviation of Gaussian distribution. When applied in a two-dimensional space, this formula produces a surface with contours as concentric circles with a Gaussian distribution from the centre point. This will result in a blur preserving the boundaries and edges better than any other more uniform bluring filters. Pupil location is identified using the Daugman's algorithm. The proposed effective integrodifferential operator for determining the centre of the pupil as well as it's radius is given by,

$$\max_{(r,\, x0,\, y0)} \left| G_\sigma(r) * \frac{\partial}{\partial r} \oint_{r,x0,y0} \frac{I(x,y)}{2\pi r} ds \right|$$

Here, $G_\sigma(r)$ represents a Gaussian Filter for the image that has been scaled to σ size. It is the smoothing function which is blurred at a scale set by σ, searching continuosly for the maximal contour integral derivative at consecutively finer scales of analysis through three parameter space of center pixel coordinates and radius values.

## 3. ALGORITHM OF LEG TRACKING MODULE

### 3.1. Selection of target

Selection of relevant target infront of the navigating system is one important section that has to be addressed in this assistance system. This can be done by a composite approach in which initially the run time frames of the video are subject to background subtraction method and then to optimal change detection. In background subtraction, the consecutive frames are subtracted with each other. This would give a black image condition to no variations infront of capturing device. Now, when the target arrives,the pixel intensity variations in the corresponding frames can be observed. In order to detect the target among these diverse features, an optimal change detection algorithm is employed which have less computation and complexity factors. These factors are essential keeping in mind, the processor capacity and ability of open sourse software that are used. The optimal change detection algorithm will provide us with required coordinates of the intial window that will be the input to the further module.

The resultant image obtained from the background subtraction technique mentioned in the previous section is given by,

$$\varphi(\mu,\vartheta,\tau) = \begin{cases} 255; & J(\mu,\vartheta,\tau) \neq 0 \\ 0; & J(\mu,\vartheta,\tau) = 0 \end{cases}$$

Where, $J(\mu,\vartheta,\tau)$ is the image obtained by the subraction of the previous frame from the current frame.

Now, the resultant image $\varphi(\mu,\vartheta,\tau)$ is left with the pixel values 0 and 255 in which the pixels of the image left with the value of 255 depict a change/movement that has occurred from the previous frame. To identify the portion bearing optimal change over previous frame, a new image $S(x,y)$ is generated.

The pixel intensity value of image $S(x,y)$ is a function of the pixel intensities of $\varphi(\mu,\vartheta)$ with in an appropriate 40x40 pixel zones.

$$S(x,y) = \frac{1}{1600} * [\sum_{\vartheta=40*x}^{40*x+39} \sum_{\vartheta=40*y}^{40*y+39} \varphi(\mu,\vartheta)]$$

Here, the size of S will be approximately 97.5% smaller compared to the size of $\varphi$. S(x, y) will be holding the information corresponding to the variations in target zones. To select the initial window of the tracking module, the coordinates of S bearing the maximum value is selected and this be $(x_0, y_0)$. New coordinates $x_1$ and $y_1$ are calculated.

$$x_1 = x_0 + 40 \text{ and } y_1 = y_0 + 40$$

The required coordinates for the tracking module will be $(x_0, y_0)$, $(x_0, y_1)$, $(x_1, y_0)$, $(x_1, y_1)$ which form a rectangular window.

### 3.2 Tracking the target

The rectangular coordinates obtained in section 3.1 provide us with intial search window for Mean Shift algorithm. Now as mentioned in key concepts section, A probability image will be generated on the window and corresponding moments are calculated. These obtained moments give the centroid (mean) of the distibution in HSV colour space. The results thus obtained are then provided to CamShift process.

$$x_c = \frac{M_{10}}{M_{00}} \qquad y_c = \frac{M_{01}}{M_{00}}$$

$(x_c, y_c)$ is the position of mean of the search window. Now, The first and second order moments are calculated as mentioned in the equations. Now, the orientation of the major axis $\theta$ and the equivalent size (length(l1) and width(l2) ) of the distribution tracked are given by

$$\theta = \frac{1}{2}\tan^{-1}(\frac{b}{a-c})$$

$$l1 = \sqrt{\frac{(a+c) + \sqrt{b^2 + (a-c)^2}}{2}}$$

$$l2 = \sqrt{\frac{(a+c) - \sqrt{b^2 + (a-c)^2}}{2}}$$

where a,b,c are the parameters calculated using first and second order moments which are given in equations provided in section 3.1

## 4. ALGORITHM OF EYE TRACKING MODULE

### 4.1 Identifying the eye portion in a given frame

First phase of the eye tracking module requires identification of the pixel coordinates corresponding to the eye portion in a given frame. For this, the Haar classification cascades for face and eye detection (discussed in the section 2.3) are utilized returning the pixel coordinates corresponding to the eye. As a result, we'll end up with a cropped eye portion for each and every frame. As the eye portion retrieval is performed for a frame taken at a long shot, usage of a camera with atleast 5MP resolution is is highly recommended so that a significant number of pixels will be holding the pupil information. This is very much essential while identifying the pupil centre and radius using the Daugman's algorithm.

### 4.2 Identification of pupil centre and it's radius

From the cropped image portion, the next step would be extracting of the centre pixel coordinates of human pupil. This plays a vital role in generation of binaries with respect to the eye movement. This is achieved by utilizing the Daugman's iris localization technique (discussed in section 2.4). Daugman's integrodifferential operator will be useful in extracting both the iris centre as well as the contour. Here, the information corresponding to the pupil's centre coordinates and it's radius are utilized for further processing.

$$\max_{(r,\, x0,\, y0)} |\, G_\sigma(r) * \frac{\partial}{\partial r} \oint_{r,x0,y0} \frac{I(x,y)}{2\pi r} ds \,|$$

Here, $x_0$, $y_0$ are the centre pixel coordinates of the pupil while r corresponds to the radius of the pupil.

### 4.3 Iris movement tracking and binaries generation

The pixel coordinates $(x_0, y_0)$ obtained from the previous phase will be useful for tracking the iris movement. For this, the initial size of cropped eye portion must be taken into consideration. Let the size of the cropped window be PxQ. Now this image is divided into three different portions. The first portion will have the pixel coordinate ranging from 0 to 2Q/5 in x-direction. For the second portion, pixel coordinate ranges from 2Q/5 to 3Q/5 in x-direction. For the the third region, pixel coordinate ranges from 3Q/5 to Q in x-direction.

Depending upon the location of pupil's centre, movement of wheel chair is initiated. For instance, if the centre lies in the region 1, appropriate digital signals are initiated to drive the wheelchair in thel left direction. The same follows for the regions 2 & 3. Region 2 corresponds to the straight movement while region 3 aids the wheel chair in moving towards right. In order to stop the wheel chair's movement, the radius of the pupil returned from the Daugman's integrodifferential operator is taken into consideration.

VOL. 11, NO. 15, AUGUST 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

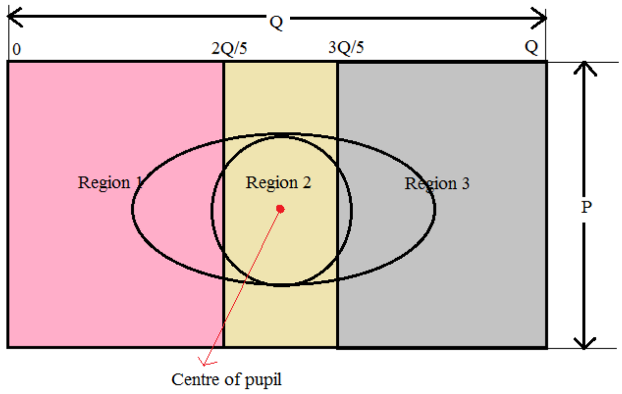


**Figure-5.** Virtual boundary declaration applied for the cropped eye portion. Region 1 corresponds to left movement, Region 2 corresponds to straight/back movement and Region 3 corresponds to the right movement.

If the radius falls below certain pre defined threshold, the movement of wheel chair comes to halt. For this, the person using the wheelchair must be closing his/her eye lid. This way, a person shall be able to control the wheelchair's movement when there's no one to assist.

## 5. RESULTS

### 5.1 Prototype of proposed leg-tracking wheel chair

**Figure-6.** Prototype of wheel chair designed for the purpose of leg tracking.

### 5.2 Modular outputs for leg tracking


**Figure-7.** Leg tracking module of mobility assistance system.

In Figure-7, the results corresponding to the leg tracking module of the proposed Mobility Assistance System are presented. For understanding purpose, the text function of OpenCV is utilized to print the direction of assisting person on the screen. As per the movement, appropriate digital signals are generated which are fed to motor drivers of the prototype used. As mentioned earlier in Section 2.2, the dimensions of newly formed elliptical window (lengths of major and minor axes) are calculated. These are used for predicting the distance of the person from the camera as the distance between them is inversely proportional to lengths of axes. From this, when the lengths of axes fall below certain pre defined threshold, stop interrupt is generated and the movement of wheel chair halts. This prevents the collision of wheel chair with assisting person.

### 5.3 Modular outputs for eye tracking

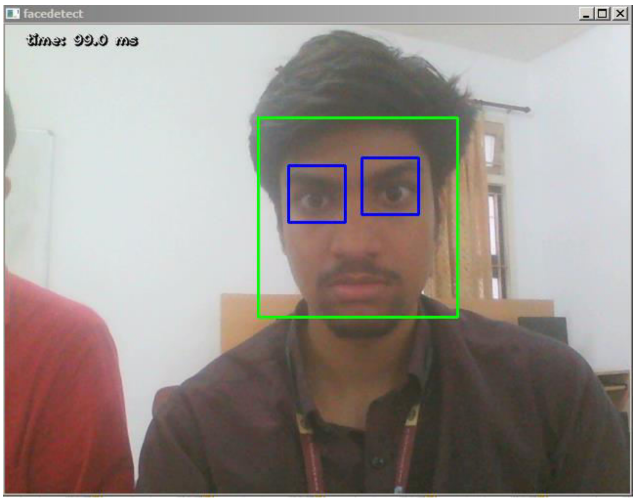


**Figure-8.** Resultant image obtained using Haar classification cascades.

By using the Haar classification cascade for face and eye, the coordinates corresponding to the left and right eye are obtined. For better understanding purpose, drawing

VOL. 11, NO. 15, AUGUST 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences



functions of OpenCV are utilized and a rectangle is drawn by taking into account the top left and bottom right coordinates of the right eye.

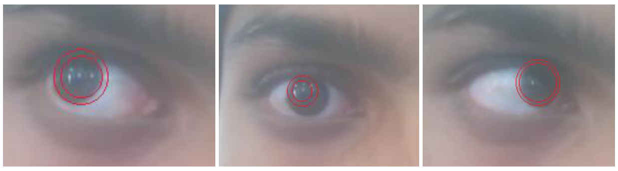

**Figure-9.** Resultant images obtained using Daugman's iris localization algorithm.

When Daugman's iris localization algorithm is applied to the cropped eye portion, the coordinates of pupil centre as well as its radis are obtained. Based on the position of the centre coordinates, relevant digital signal is supplied to the motor drivers so as to aid the wheelchair's movement in the approporiate direction. When the radius returned from the Daugman's integrodifferential operator is less than certain threshold, the wheelchair's movement comes to halt. This happens when the person using the wheel chair closes his eyes.

## 6. CONCLUSIONS

The current system has leg tracking and eye tracking modules which deploys image processing algorithms like Background Subtraction, Camshift, Haar Classification Cascades, Daugman's Iris localisation which makes the system advanced and enables physical assistance for 1.6% of population of India who have the disability in movement and seeing. The cascade of background subtraction and CamShift algorithms result in automatic tracking of legs of assistive person unlike the conventional tracking systems in which the initial window coordinates are be pre-declared. The pupil detection module of the proposed assistive system doesn't require a close shot of eye as proposed in various other eye-tracking based wheelchairs. Haar eye classification cascade will automatically return the coordinates of eye portion from a long shot image making the proposed assistive system unique.

## REFERENCES

[1] http://censusindia.gov.in/Census_And_You/disabled_population.aspx

[2] Hori J., Sakano K. and Saitoh Y. 2006. 'Development of a communication support device controlled by eye movements and voluntary eye blink. IEICE Transactions on Information and Systems, E89-D. (6): 1790-1797.

[3] K. T. V. Grattan, A. W. Palmer and S. R. Sorrell. 1986. Communication by Eye Closure-A Microcomputer-Based System for the Disabled. IEEE Transactions on Biomedical Engineering. BME-33(10).

[4] Q.X. Nguyen and S. Jo. 2012. Electric wheelchair control using head pose free eye-gaze tracker. Electronics Letters. 48(13).

[5] Rory A. Cooper. 1995. Intelligent Control of Power Wheelchairs. IEEE Engineering in medicine and Biology, 0739-51 75/95.

[6] Djoko Purwanto, Ronny Mardiyanto, Kohei Arai. 2009. Electric wheelchair control with gaze direction and eye blinking. Artif Life Robotics. 14: 397-400.

[7] J.G.Daugman. 2004. How iris recognition works. IEEE Trans. Circuits and Syst. for video Tech. 14(1): 21-30.

[8] Sibert LE, Jacob RJK. 2000. Evaluation of eye gaze interaction. In: Proceedings of the SIGCHI conference on Human Factors in Computing Systems. New York, NY, USA: ACM, CHI '00. pp. 281-288.

[9] Society AE. 1991. American electroencephalographic society guidelines for standard electrode position nomenclature. Journal of clinical neurophysiology. 8: 200-202.

[10] Chen Y, Newman W. 2004. A human-robot interface based on electrooculography. In: Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on. 1: 243-248.

[11] G. Andrews, R. Askey and R. Roy. 1999. Special Functions. Cambridge University Press, New-York, USA.

[12] A. Erdely and all. 1954. Tables of integral transforms. Vol.II, Mc Graw-Hill Book Compagny, New-York, USA.

[13] S. Mitra, T. Acharya: 2007. Gesture Recognition: A Survey. IEEE Transactions on Systems, Man, and Cybernetics. Part C: Applications and Reviews. 37(3): 311-324.

[14] Er. Vijay Dhir and Dr. (Ms) Maitryee Dutta. New Method of IRIS RECOGNITION Based on J. Daugman's Pinciple. Second International Conference on Emerging Trends in Engineering and Technology, ICETET-09.

[15] Barea R, Boquete L, Mazo M, López E. 2002. Wheelchair guidance strategies using eog. Journal of Intelligent and Robotic Systems. 34: 279-299.

[16] Locked-In Syndrome Information Page: National Institute of Neurological Disorders and Stroke (NINDS). 2007. <http://www.ninds.nih.gov/disorders/lockedinsyndrome.htm>.

[17] Paul Viola, Micheal Jones. 2001. Rapid Object Detection using a Boosted Cascade of Simple Features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. CVPR.

[18] G. bradski, and R. Gary, and L.M. Olga. 1998. Computer Vision Face Tracking For Use in a Perceptual User Interface. Intel Technology Journal Q2'98. 10(15): 1-15.

[19] G. Bradski, T. Ogiuchi, and M. Higashikubo. 2010. Visual Tracking Algorithm using Pixel-Pair Feature. International Conference on Pattern Recognition. (4): 1808-1811.

[20] E. David, and B. Erich, and K. Daniel, and S. Anselm. 2010. Fast and Robust Camshift Tracking. IEEE Transaction. (8): 1-8.

[21] Intel Corporation. 2001: Open Source Computer Vision Library Reference Manual. 123456-001.

[22] Cheng Y. 1995. Mean shift, mode seeking, and clustering. IEEE Trans. Pattern Anal. Machine Intel. 17: 790-799.

[23] Comaniciu D., Ramesh V. and Meer P: 2003: Kernel Based Object Tracking, IEEE Trans. Pattern Analysis Machine Intell. 25(5): 564-575.