

# Gaze Direction based Mobile Application for Quadriplegia Wheelchair Control System

Dr. Muayad Sadik Croock<sup>1</sup>, Dr. Salih Al-Qaraawi<sup>2</sup>, Rawan Ali Taban<sup>3</sup>

Computer Engineering Department  
University of Technology  
Baghdad, Iraq

**Abstract**—People with quadriplegia recruit the interest of researchers in introducing automated movement systems for adopted special purpose wheelchairs. These systems were introduced for easing the movement of such type of disabled people independently. This paper proposed a comprehensive control system that can control the movement of Quadriplegia wheelchairs using gaze direction and blink detection. The presented system includes two main parts. The first part consists of a smartphone that applies the propose gaze direction detection based mobile application. It then sends the direction commands to the second part via Wi-Fi connection. The second part is a prototype representing the quadriplegia wheelchair that contains robotic car (two-wheel driving car), Raspberry Pi III and ultrasound sensors. The gaze direction commands, sent from the smartphone, are received by the Raspberry Pi for processing and producing the control signals. The ultrasound sensors are fixed at the front and back of the car for performing the emergency stop when obstacles are detected. The proposed system is based on gaze tracking and direction detection without the requirement of calibration with additional sensors and instruments. The obtained results show the superior performance of the proposed system that proves the claim of authors. The accuracy ratio is ranged between 66% and 82% depending on the environment (indoor and outdoor) and surrounding lighting as well as the smart phone type.

**Keywords**—Gaze direction detection; mobile application; obstacle detection; quadriplegia; Raspberry Pi microcomputer

## I. INTRODUCTION

Paralysis is most often caused by damage in the nervous system, especially the spinal cord injuries as results of a traffic accident or from some diseases. These diseases can include cerebral palsy or multiple sclerosis (MS) which lead to quadriplegia problems. Quadriplegia typically occurs as a result of an injury at the thoracic spinal levels (T1 or above) causing a loss of sensation and movement in all four limbs [1]-[2].

Paralyzed people are in need of an assistant to move them in their wheelchair, therefore they often feel powerless and burden on others [3]. Consequently, it became important to designing and developing a semi-automatic wheelchair, wherein the movement of the wheelchair can be controlled by movements of the different organs of the human body [1].

Different interfaces have been developed for dealing with people suffering from different disabilities; such as joystick control, hand control, head control and voice control. Other

systems employ infrared head-pointing devices, mouth-actuated joysticks, tongue-movement analysis, or head-mounted cameras as interfaces [4].

Communication abilities for severe quadriplegic people are practically limited to "Yes" and "No" responses using eye movements or blinks because the only muscles around the eye can be fully and easily controlled by them [5]. Thus, not all people are able to move around using the above-mentioned interfaces.

Currently, eye tracking technology has been used to design wheelchair control systems and other useful communication systems to help disabled people. Different techniques for eye tracking are developed such as Electrooculography (EOG), search coil method or Infrared oculography using head-mounted cameras. Most of the methods are considered to be uncomfortable as they have a drawback of using attached devices to the user's face [5], [6].

The aim of this paper is to design and implement a system to help quadriplegic people to be self-reliance in driving their wheelchair without being a burden on others. Therefore, an affordable eye tracking based control system for driving a wheelchair prototype model is presented. The system adopts the front camera of the mobile phone as a sensor to capture a real-time video for further processing using a designed mobile application that detects and tracks the eye movements. The detected action is sent as a command to a robotic car with two motors as a prototype instead of the quadriplegic wheelchair due to lack of availability. This is to present the move maneuvers, such as forward, backward, right, and left or fully-stop it, as shown in Fig. 1.

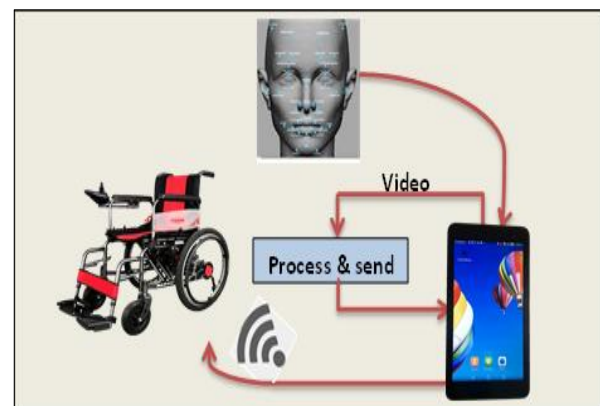


Fig. 1. Basic block diagram.

This paper is organized as follows. Section two presents the related works with wheelchair controlling systems. Different techniques for controlling the wheelchair are described in this section too. The proposed prototype control system based on eye-movement tracking and blink detection is shown in section three. Tests and experimental results are introduced in Section four. Section five and six give the limitation and a brief conclusion about the overall system respectively.

## II. RELATED WORKS

There are many types of research concerned with designing control systems to help physically disabled people in driving their electric wheelchair.

In 2014, Biswajeet Champaty and et al. [7] proposed a prototype of an electric wheelchair control system based on Electrooculography (EOG) signals to move it in different directions. The system includes the blink detection in addition to gaze direction detection. The system was considered to be uncomfortable because of the face-attached EOG electrodes.

In 2017, Rana F. and Hani S. [8] presented a head direction based control system for disabled people to help them drive their wheelchair. The system used Viola Jones algorithm to detect four head directions. MATLAB is used for processing and sending the signal to the Arduino microprocessor for control the movement of a designed prototype. This system cannot be used by some people with severe paralysis.

In 2017, Xiang Gao and Lei Shi [9] designed a control system based on hand gesture detection for a robotic wheelchair to help the aged and the disabled people. The hand motion was detected using the Kinect which is a line of motion sensing input devices. The processing operations are performed on a PC placed in front of the user and then sending them to a DSP 28335 microcontroller to control the wheelchair. This system can be used only by Paraplegic and aged people because it needs a fully controlled hand.

In 2018, Raju Veerati, et al. [1] used a webcam with IR illuminators to track eye gaze direction with the help of MATLAB environment for processing. Detected gaze directions are used to control a wheelchair prototype which occupied by an ultrasonic sensor for obstacles avoidance to help people with disabilities. The results proved that the processing operations are quite slow; however the system runs with good accuracy. There are many types of research

concerned with designing control systems to help physically disabled people in driving their electric wheelchair.

## III. PROPOSED SYSTEM

In general, the proposed system's structure consists of two main parts: the eye-sight placed mobile phone and the mini electric car which is used as a prototype of the quadriplegic wheelchair. The user's eye is tracked to estimate different actions and use it as input technique to the mobile. Hence, this application replaces the need of button pressing by finger touching. Based on the estimated eye actions, a related command is transferred via Wi-Fi connection to a microcomputer attached to the car for real-time motors controlling.

Different tools have been used to accomplish the work on the introduced system including:

### *The hardware tools:*

- An Android smartphone.
- Two-wheel drive (2WD) electric car.
- Raspberry Pi microcomputer.
- Two ultrasonic sensors.
- Step down voltage regulator.
- Battery.
- Breadboard and connectors.

### *The software tools:*

- Android Studio (2.3.3) IDE.
- OpenCV (2.4.9) library.

This section can be divided into three sub-sections for easing the reading flow of the presented work.

### *A. Proposed Algorithm*

Most of the processing operations are performed at the software side which is the mobile application. In addition, the hardware side (the microcomputer) is responsible for generating the controlling signals used for car driving. Fig. 2 shows the proposed algorithm as a flowchart. All the detection operations are run in the background of the application which are the face, eye and pupil detection.

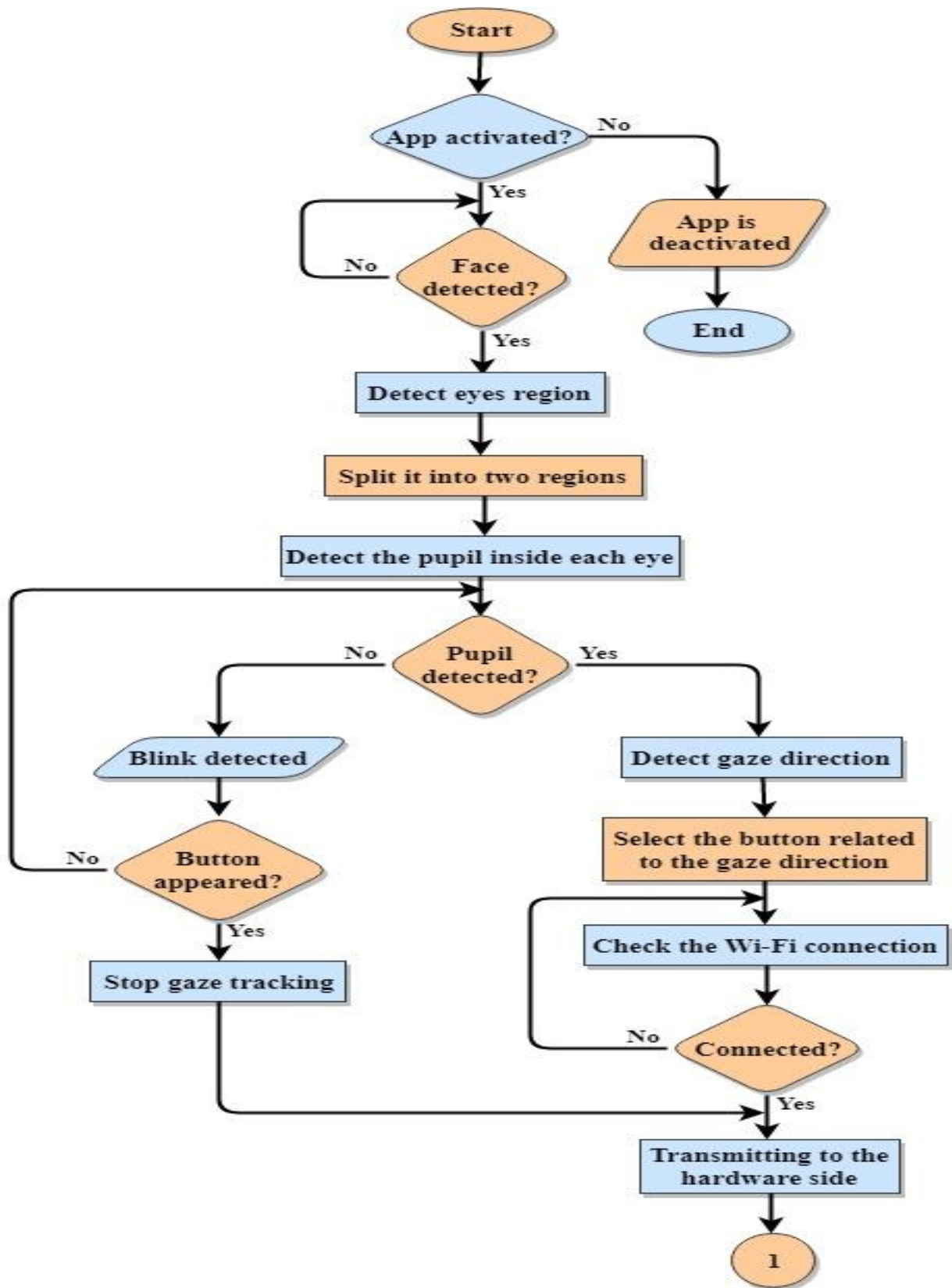


Fig. 2. (a) The software side.

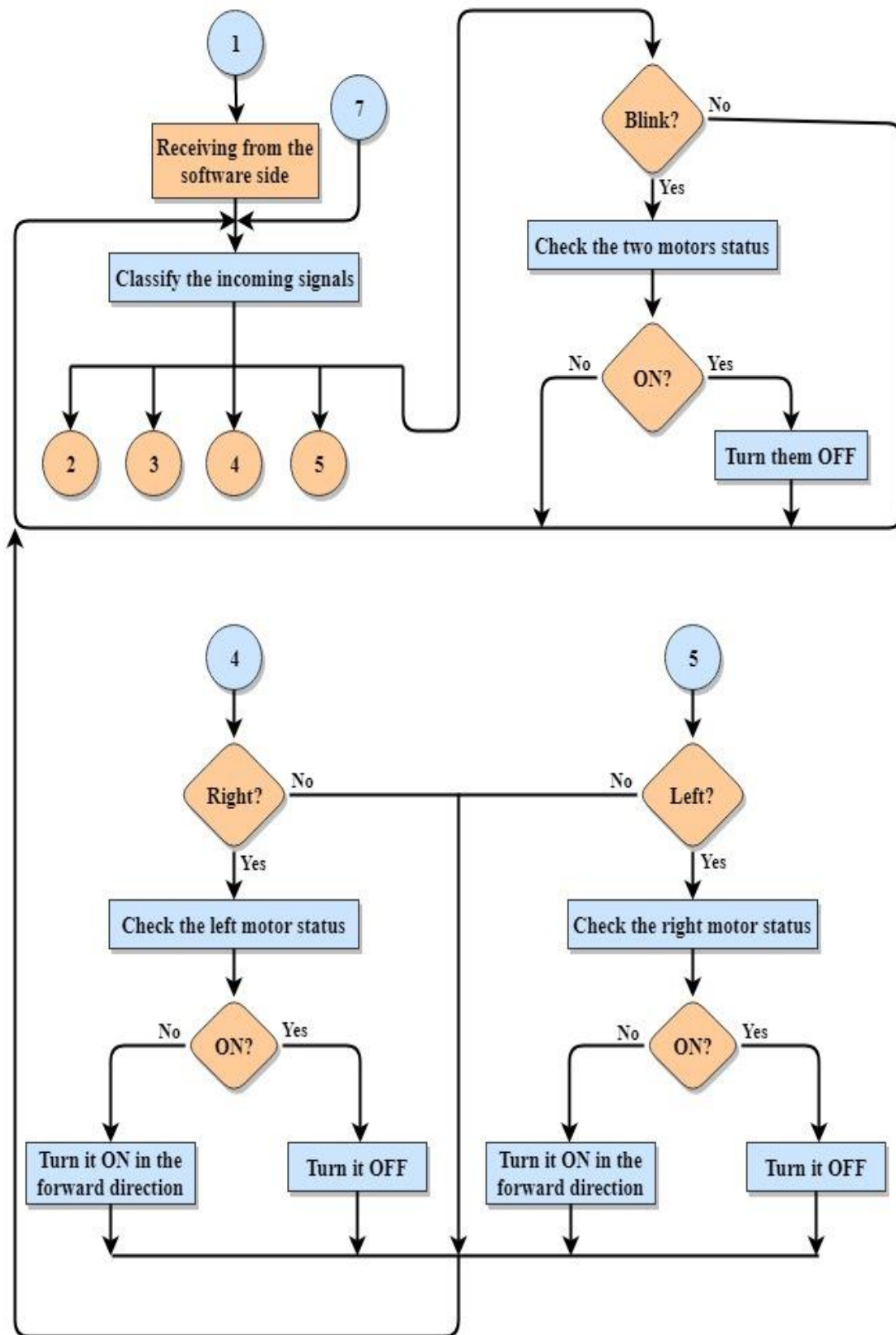


Fig. 2. (b) The hardware side: the blink, right and left gaze directions responses.

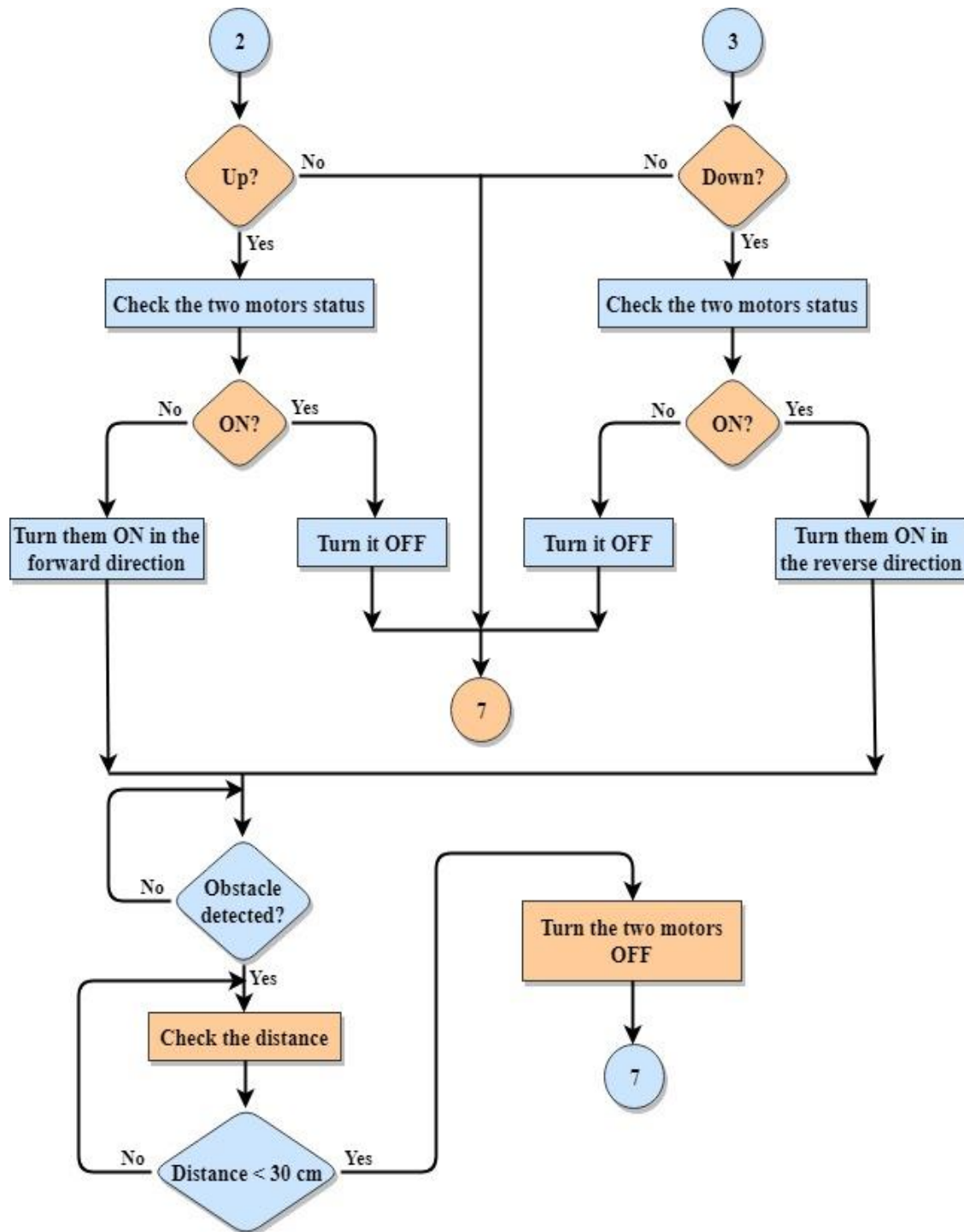


Fig. 2. (c) The hardware side: the up and down gaze directions responses.

The application can detect two types of eye actions including the gaze direction detection and eye blinking detection which then send using Wi-Fi connection to the hardware side for controlling purposes. The "right" and "left" gaze directions used for turning the car to right or left respectively Whereas the "up" and "down" gaze directions used for moving the car in forward or backward direction respectively. The blinking action is used for eye-tracking and car-controlling deactivating. Two ultrasonic sensors have been

used for obstacles detection during forward and backward car movements.

### B. The software side

It is the core part of the proposed system which is a mobile application developed to track the human eye and detect different actions from it. The designed application interface is depicted in Fig. 3.





Fig. 3. The application user interface.

The application is designed under the android studio IDE to be applied on android based smartphones and tabs. A machine learning based algorithm which is viola-jones algorithm has been adopted in this application processing. It is an algorithm used to train a classifier on numerous positive and negative images. It is used to detect a specific object in these images by applying scanning process several times at different scales to detect different sizes of the object [10].

At the other hand, Haar-like features based cascade classifiers have been used because of the short time needed to compute an extensive set of features with efficient results of detection related to using this classifier [11]. As aforementioned, working on the application image processing is accomplished using the following stages respectively:

1) *Face detection stage*: A trained Haar cascade classifier for face detection has been used in this stage. Wherein, the human face is detected and surrounded by a rectangle in real-time video as shown in Fig. 4(a).

2) *ROI calculation stage*: In this stage, the Region of Interest (ROI), the eyes-region, is approximately determined by calculating the coordinates of the upper left corner of the eye region (x, y), the width and the height depending on the dimensions of the detected face rectangle by using (1), (2) and (3), respectively, assuming that the region of eyes occupies 1/3 of the face height and 3/4 of the face width [12]:

$$EyeArea_{(x,y)} = (FR.x + \frac{FR.w}{8}, FR.y + \frac{FR.h}{4.5}) \quad (1)$$

$$EyeArea_{width} = \frac{3}{4} FR.w \quad (2)$$

$$EyeArea_{height} = \frac{FR.h}{3} \quad (3)$$

Thus, the ROI can be determined and located as shown in Fig. 4(b). The next step is to split the resulted eye region into two regions as shown in Fig. 4(c). This is for distinct the two eyes from each other and shrink the ROI as a consequence, easing the detection of both eyes separately in the next stage. This is accomplished by following (4), (5) and (6) [12]:

$$RightEye_{(x,y)} = (FR.x + \frac{FR.w}{2}, FR.y + \frac{FR.h}{4.5}) \quad (4)$$

$$RightEye_{width} = \frac{7}{16} FR.w \quad (5)$$

$$LeftEye_{(x,y)} = (FR.x + \frac{FR.w}{16}, FR.y + \frac{FR.h}{4.5}) \quad (6)$$

Where, FR is the face rectangle upper left corner point. Moreover, w and h are the width and the height of the rectangle respectively. The width of the left eye is computed using (5), whereas the heights of both eyes are separately computed using (3).

3) *Pupil detection stage*: The proposed algorithm applies another Haar cascade classifiers on the two regions, detected in Fig. 4(c), for separately eyes detection. The right and left eyes are detected without the eye-brow and the unwanted skin regions [12]. After that, the pupil's point is obtained which is the darkest point in that region. Then, it is surrounded by a circle to distinguish and track it as shown in Fig. 4(d).

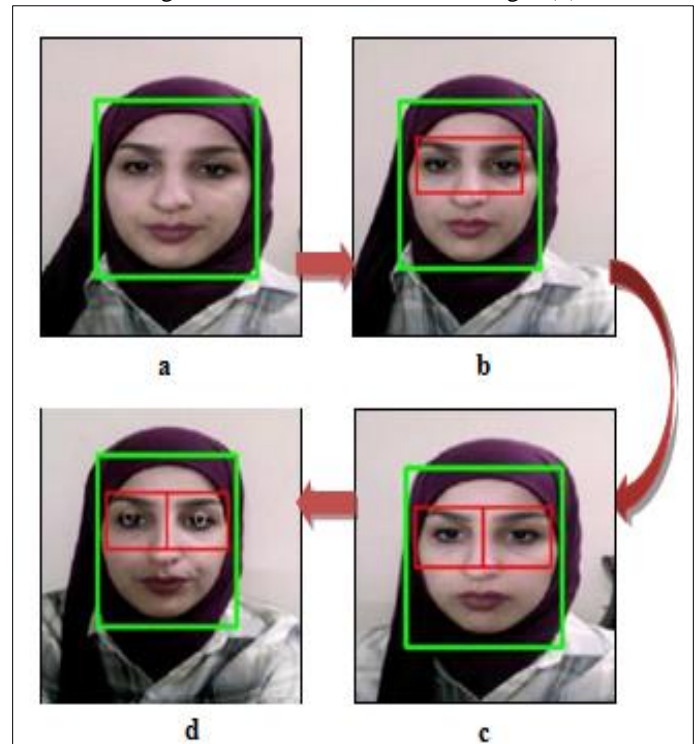


Fig. 4. The first three stages of processing: (a) is the first stage, (b-c) for the second stage and (d) is the third stage.

4) *Eye-actions detection*: It is the final stage in the application processing. The two eyes regions detected in Fig. 4(c) are separately divided into a 5x5 grid, as depicted in Fig. 5, to use them as thresholds of the pupil's movement directions range, used in detecting the following two actions:

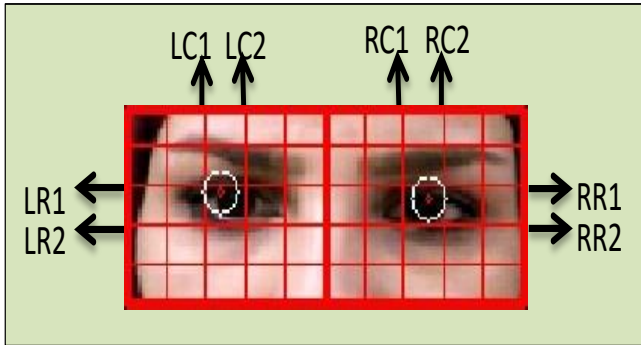


Fig. 5. The division of both eyes regions into a 5x5 grid.

a) *Gaze direction detection*: Gaze direction detection: Four eye gaze directions have been detected at this stage including up, down, right and left. This is accomplished by depending on the taken thresholds of the right eye solely, as expressed in the pseudo code of Fig. 6. Ultimately, a specific button is pressed after detect its related gaze direction.

b) *Blink detection*: The absence of the two eyes pupils means that the eye blinking action is performed. Nonetheless, this type of actions can be obtained involuntarily by the user. Therefore, this action is detected under the condition of the non-existence of pupil circles of both eyes with some conditions that can be summarized as:

- The user's gaze must be at the forward position first before blinking, to avoid the overlap with the gaze directions action.
- The user's eye must be continually closed for 2sec.
- The blink must be hard enough to guarantee the eyelashes disappearance. This is to avoid overlapping with the condition of detecting the pupil depending on

the darkest point which is the eyelashes in the case of the closed eye.

```

/*detect the eye gaze direction depending on the right eye*/
Let: RP = right eye pupil
    LP = left eye pupil
    RPC = right pupil circle's center
    GD = gaze direction
/*in addition to the rows and columns names mentioned in figure (5) */
If RP && LP are detected then
  If RPC >= RC2 then
    GD = "right direction"
  elseif RPC <= RC1 then
    GD = "left direction"
  elseif RPC <= RR1 then
    GD = "up direction"
  elseif RPC >= RR2 then
    GD = "down direction"
endif
endif
endif
endif
    
```

Fig. 6. Pseudo code of gaze direction detection method.

C. *The hardware side*:

This part mainly consists of a robot car that is controlled using the attached Raspberry Pi microcomputer. This section can be divided into three sub-sections as follows:

1) *Car aggregation and setup*: As illustrated in Fig. 7, a two-wheel motor driver (2WD) robotic car is adopted in this system with other hardware peripherals that can be listed as follows:

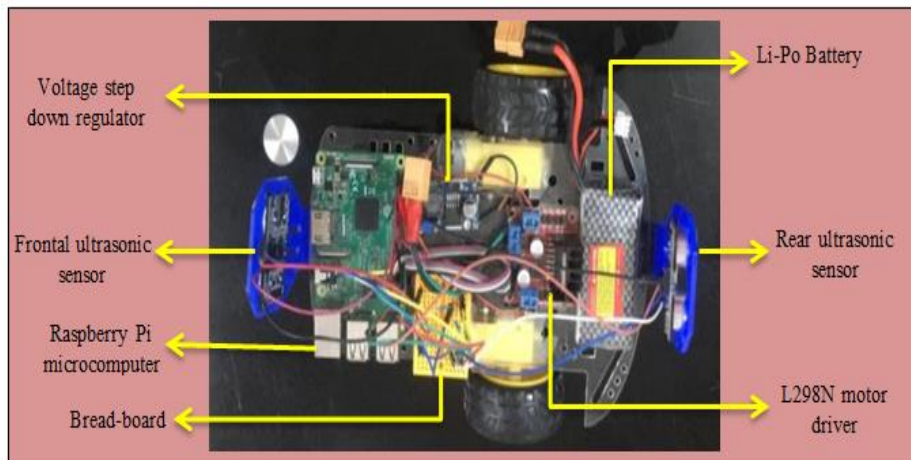


Fig. 7. Car module hardware requirements.

a) *The Raspberry Pi III model B microcomputer:* As shown in Fig. 8, it is a small, credit card-sized computer developed in the United Kingdom by the Raspberry Pi Foundation. The module is well equipped with different kinds of input/output ports and it performs quite well in the tasks that are related to the proposed system. It is used to control the movement of the car's motors [13].



Fig. 8. Raspberry Pi III model B.

b) *The L298N motor driver:* It is a high current, high voltage dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors [14]. Fig. 9 shows the adopted motor driver model.



Fig. 9. L298N motor driver.

It is used in this system to drive the two-car motors in different directions. Li-Po battery is used as a power supply for the motor driver.

c) *The ultrasonic sensors:* For obstacles detection, two ultrasonic sensors are used one in the front and back of the car.

d) *The voltage step down regulator:* It is used to supply the Raspberry Pi with its required voltage from the motor driver. The used model of this regulator is depicted in Fig. 10.



Fig. 10. Voltage step down regulator.

2) *Controlling process:* The central part of controlling in this system is the Raspberry Pi. Its built-in Wi-Fi communication feature is exploited, for establishing a Wi-Fi connection to be used as an access point, connected to the mobile phone. Raspberry Pi receives the command of the

detected eye action from the software side and then controls the movement of the car using the motor driver.

The car can move forward and backward as well as turning left and right depending on the incoming command of the detected gaze direction. Receiving the same command twice respectively, regardless of the time period between the two commands, means that the first one is for moving toward the intended direction and the second one is to stop this movement.

In addition, two different stopping commands can be applied to the car. The first one terminates the current movement direction of the car which accomplished by receiving the same command twice. The second one stops the car which accomplished in the case of blinking receiving.

3) *Obstacles avoidance:* As mentioned above, two ultrasonic sensors are used to detect the obstacles by measuring the distance between the car and the faced obstacles. The Raspberry Pi fully stops the car if the measured distance is below than (30 cm) to be waiting until the obstacles are removed or the user sends another command. The system can detect the obstacles during the forward and backward directions movements.

#### IV. EXPERIMENTAL RESULTS

Different experiments are conducted to the determination of the limits, rules and some guides for using the introduced system. Because the proposed system deals with a camera as a sensor, one of the important factors to be test is the intensity of light. In addition, the distance between the user's eye and the camera is another important factor that must be checked and determined. This is due to the architecture of the system that deals with the pixels positions, to determine the gaze direction, which affected by this distance.

Before illustrating the adopted experiments, a major point is necessary to be stated which is the patient's head position. It needs to be fixed straight forward in front of the device's camera to obtain perfect detection accuracy. In addition to that the device is also must be fixed by an adjustable holder.

The proposed system was tested and evaluated on Huawei T1-701u tablet with Quad core 1.2 GHz, 2MP camera and 600x1024 resolution which running Android 4.4.2 version.

Different cases has been studied and tested to check the effect of both the light and the distance factors which can be concluded as follows:

- The distance between the patient's eye and the device's camera: To determine the optimal distance between the eye and the camera, different values for this distance are supposed and applied on the system with no change on the ambient light intensity which is the room's florescence lamps. The florescence lamps are not directly faced the user's eye nevertheless it already placed near the room's roof.

##### *Case 1: 45cm, florescence lamps*

In this case, 45cm is the distance between the patient's eye and the device's front camera under only the florescence lamps.



The obtained overall accuracy is 52%. The blink detection was better than the gaze direction detection because the gaze direction detection depends on the difference between pixels' positions which vary depending on the distance between the face and the camera.

*Case 2: 50cm, florescence lamps*

In this case, the distance is increased by 5cm amongst the first case with maintaining the same light intensity of the first case. In this experiment, the obtained accuracy is 78%. Wherein, the gaze direction detection and the blinking detection dependently give acceptable accuracy.

*Case 3: 55cm, florescence lamps*

Due to the increased distance between the camera and the user's eye, and hence the difficulty of detecting the pupil point, the obtained accuracy of this experiment was 62% which is lower than the accuracy of the previous experiment.

- The intensity of the ambient light: The determining of the intensity of the ambient light during using the system is another challenge in the system test. To do that, a table lamp with 2 levels of intensities, adjusted manually, is used. From the previous experiments, because the optimal accuracy is obtained when the distance is 50cm, it was considered as the optimal distance and used as a constant factor during testing the

intensity of light effect. The lamp is placed in front of the user's face above the device to determine the effect of the high intensity of light when it directed to the eye.

*Case 1: 50cm, (florescence lamps and table lamp's low intensity)*

In addition to the ambient light of room's florescence lamps, the low intensity of the table lamp is used which faces the user's face. Under the conditions of this case, the overall system accuracy is 56%.

*Case 2: 50cm, (florescence lamps and table lamp's high intensity)*

This experiment proved that as much as the intensity of light increased, the accuracy of the system will decrease. In this experiment, the system is in worst case because of the increased brightness on the pupil and hence the difficulty of detecting the required pupil.

*A. Car Locomotion Test*

As mentioned previously, the car can move in four different directions (forward, backward, right turn and left turn) depending on the user gaze direction detected at the mobile application side (software side). Fig. 11-14 express the proposed system of the four detected gaze directions and the response of the application interface with the car movement indicated by the white LED brightness.

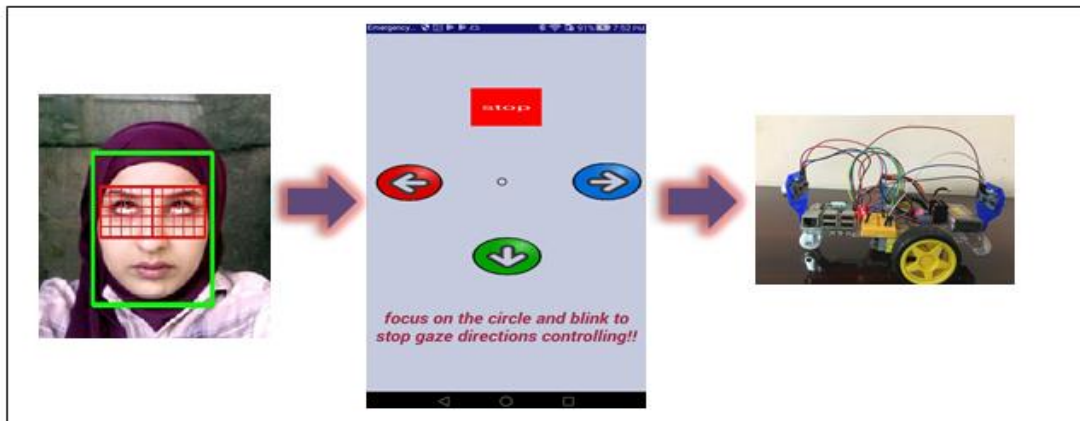


Fig. 11. Forward movement direction.

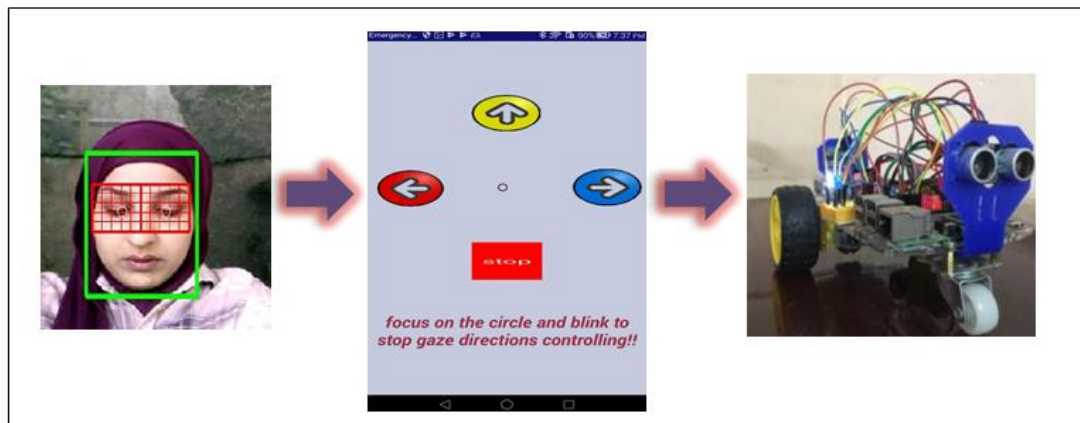


Fig. 12. Backward movement direction.

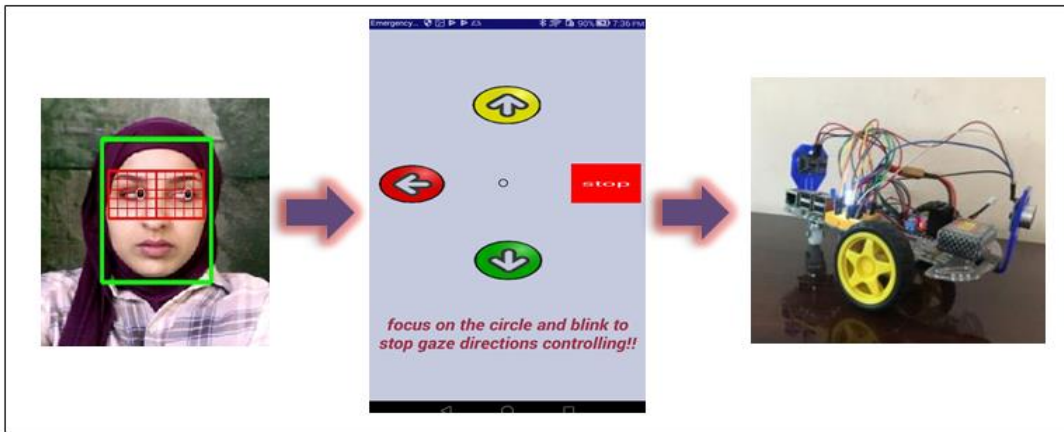


Fig. 13. Right-turn movement direction.

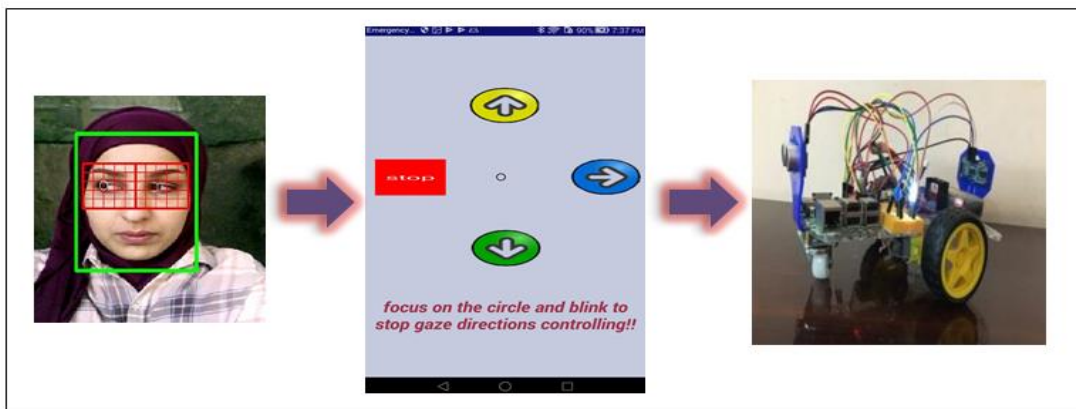


Fig. 14. Left-turn movement direction.

### B. System Deactivation Test

If the user needs to deactivate the overall system and being free in his/her eyes motion, he/she needs to use the blinking action. In this case, the system is deactivated and the car has not received any moving action until the next blinking received to reactivate the overall system. The blinking action is depicted in Fig. 15.

### C. The Obstacles Stop Test

From the experiments, the car has the ability to perfectly detect the obstacles during forward and backward movement directions using front and rear ultrasonic sensors. Fig. 16 shows the car is stopped after obstacle detection in two states forward and backward which indicated by the blue LED brightness (the white LED OFF state indicate the OFF state of directions movement).

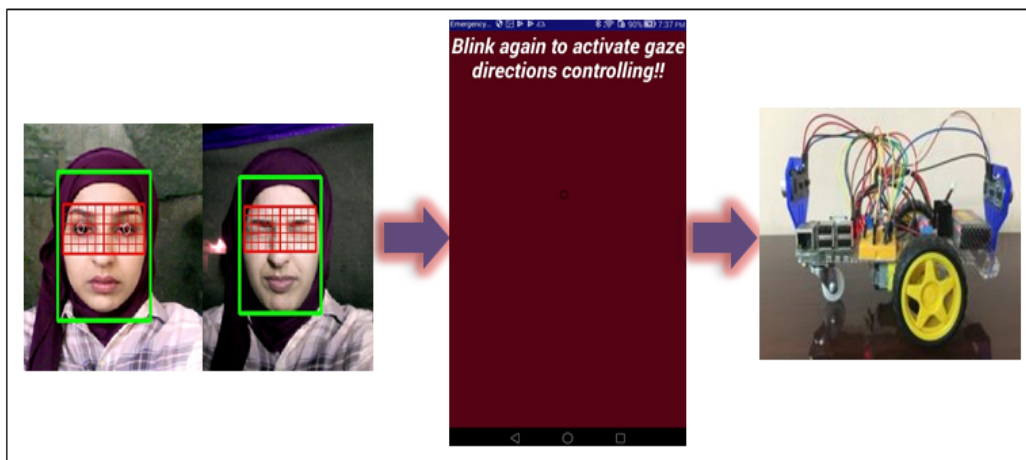


Fig. 15. Stop car movement.

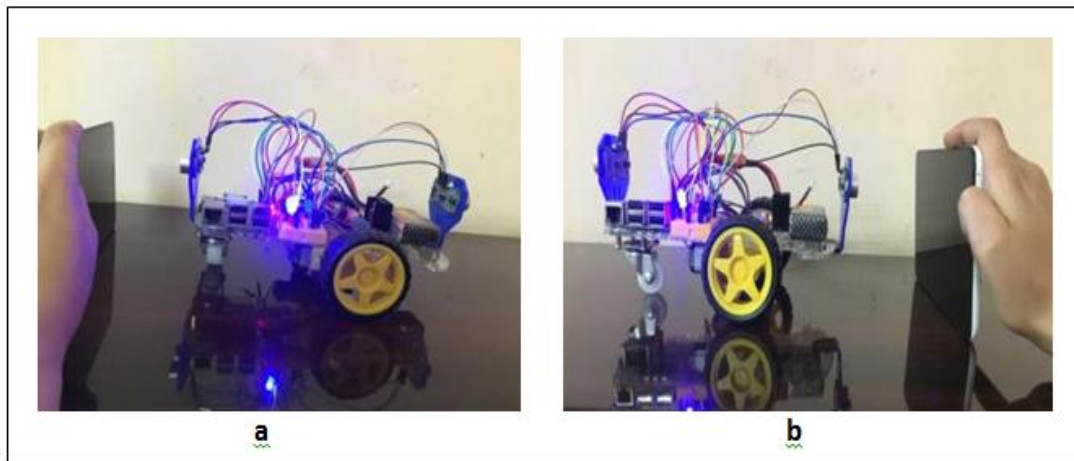


Fig. 16. Obstacle avoidance: (a) the frontal sensor and (b) the rear sensor.

#### D. Efficiency Evaluation of the Proposed System

There are different factors can impact on the system efficiency. Some of them have effects on the interface between the user's eye and the application, while others impact on the communication between the application and the car. Ultimately, all of them have effects on the system performance in direct or indirect impaction ways. As illustrated in the previously mentioned experiments, *the light intensity* is a very important factor that can affect the efficiency of the application and hence the overall system. Different light intensities have been tested during the experiments; ultimately, it is proved that the intensity of light must not be very high due to its effects on the size of pupil which causes difficult pupil detection [15]. Furthermore, the increased intensity of light can make the pupil brighter which causes incorrectly detection. This is because the pupil does not remain dark as needed. In addition, the low light intensity causes low detection accuracy because of the darkness of the shadow affected on the eye region and thus make the pupil difficult to detect. Therefore, the ambient light, during the system usage, must be neither directed to the eye nor a poor intensity light. Instead, it preferred to use a light source far from the eye and with acceptable intensity such as the florescence lamps.

Since, the efficiency of the overall system is affected by the light intensity; its accuracy can vary depending on the ambient environment (outdoor or indoor). Therefore, several experiments have been applied on the system in outdoor to test its accuracy. As illustrated in Table I, the confirmed results prove that the system accuracy is decreased in the outdoor environments due to the increased light reflected by the eye especially under the sun light. The solution for this issue can be performed by installing an umbrella above the wheelchair.

Different experiments reveal *the optimum distance* required between the user's eye and the mobile's camera which must be almost 50cm indoor and 54cm outdoor to obtain optimal eye-actions detection.

Although the system runs with accepted accuracy results under low device's camera resolution and processing speed like the Huawei phone used in the experiments of section 4.2, the high camera resolution and fast processing speed are preferred

to obtain better and perfect results. Therefore, in addition to the Huawei tab, a second device has been used through system testing which is a GALAXY tab with Quad core 1.6 GHz, 2MP camera and 800 x1280 resolution of the camera running Android 4.2 version. As shown in Table I, the obtained accuracy, using this device, outperforms the accuracy using the Huawei tab because of the powerful mobile specifications (especially its camera efficiency). Thus, *the device's processing speed* and its *camera resolution* have another impact on the system efficiency.

There are different factors can impact on the system efficiency. Some of them have effect on the interface between the user's eye and the application. Others impact on the communication between the application and the car. Ultimately, all of them have an impact on the system performance regardless of whether it was direct or indirect impact.

TABLE I. THE OBTAINED ACCURACY RESULTS.

Action	On Galaxy phone		On Huawei phone	
	Indoor accuracy	Outdoor accuracy	Indoor accuracy	Outdoor accuracy
Right gaze direction	90%	70%	90%	70%
Left gaze direction	100%	80%	90%	60%
Up gaze direction	80%	70%	70%	50%
Down gaze direction	70%	70%	70%	60%
Blinking	70%	90%	70%	90%
<b>The whole system accuracy:</b>	82%	76%	78%	66%

From this table, we can observe that the low detection accuracy ratio of the up and down directions as compared to the right and left directions is appeared because the horizontal line of the eye is wider than the vertical line. As a consequence, the horizontal movement range of the eye from left to right and vice versa is faster to detect than the vertical movement range between up and down directions. Moreover, this table also

shows a difference in overall accuracy when the device's specifications are varied. In addition, a noticeable difference is seen between the usage in outdoor and indoor environment accuracy. Wherein, the system wins accuracy in the indoor environment of about (6%-12%) as compared to the gained accuracy in the outdoor environment. The high accuracy is obtained by using the system indoor with powerful device's specifications which was almost 82% that shows the superior performance of the proposed system considered being optimal for disabled people's usage.

At the other hand, another major factor that can limit the system accuracy is *the user's head position*. The proposed system does not employ a feature for supporting the tilted head in the tracking of the eye. Therefore, the user's head must be kept straight forward in a stable manner.

*The speed of the Wi-Fi connection and the microcomputer processing speed* are other critical points in the system efficiency testing. Because the system is mainly designed for controlling aspects for a very sensitive group of the community (disabled people), it must be fast in receiving the control command and directly process it with fast latency after signal transfer without any noticeable delay. In the proposed system, the used Wi-Fi connection strategy, accomplished using socket connection makes the system active in real-time, and responsive with fast latency time between sending and receiving the command. In addition to that, the Raspberry Pi III model B is run with 1.2 GHz processor speed which is fast as compared to the old versions of related embedded systems.

#### V. CONCLUSION AND SUGGESTIONS FOR FUTURE WORK

An affordable control system based on eye-tracking technology was designed and implemented to control the movement of a mini robotic car as a prototype of the wheelchair. The system consisted of two main parts: the eye tracking application (software side) and the robotic car (hardware side). The eye tracking application was designed under an Android operating system that adopted the device's front camera as a sensor that captures a real-time video. The captured videos were processed using Viola Jones algorithm with Haar cascade classifiers to detect four gaze directions in addition to the blinking action. The resulting signals were sent via Wi-Fi to the second part of the system which is Raspberry Pi microcomputer attached to the car. Depending on the incoming signals, the Raspberry Pi controlled the movement of the used car motors. Wherein, the right and left gaze directions turned the car to the right and the left whereas up and down gaze directions moved the car forward and backward. In addition, the blink action has been used to fully stop the car. Experiments on the system have shown that it is easier to control the car using only the eye gaze directions with high performance in term of accuracy and precision. Moreover, it is important to note that the implementation cost of the proposed system was really low in comparison with manufactured similar systems. This due to the use of low cost embedded system and the proposed mobile application can work with low specifications phones or tablets.

To enhance the system by adding additional features as a future work, different suggestions are listed below:

- Making the system responsive with the people who are in black skin and wearing eye-glasses on their eyes by developing the proposed algorithm.
- Providing a feature on the application that enable it to calculate the distance between the device's camera and the user's eye and prevent running the system unless this specified distance is regulated.
- Integrating the iris recognition technology alongside with the application to obtain a full system security.
- Designing the same application using swift programming language to be able to support MAC OS based devices such as iPhones and iPads.
- Integrating a system on the wheelchair side that responsible of auto-driving the wheelchair of the patient from place to place automatically and based on information received from the improved application.

#### REFERENCES

- [1] R. Veerati, E. Suresh, A. Chakilam and S. P. Ravula, "Eye monitoring based motion controlled wheelchair for quadriplegics", *Microelectronics, Electromagnetics and Telecommunications*. Springer, Singapore, pp. 41-49, 2018.
- [2] J. A. Athanasou, "Spinal Cord Injury", *Encountering Personal Injury. Studies in Inclusive Education*, pp. 157-164, 2016.
- [3] B. Buvanswari and T. Kalpalatha Reddy, "Eye scrutinized wheel chair for people affected with tetraplegia", *International Journal of Computer Science, Engineering and Information Technology*, vol. 5, no. 2, pp. 15-24, 2015.
- [4] R. Cerejo, V. Correia and N. Pereira, "Eye controlled wheelchair based on arduino circuit", *International Journal of Technical Research and Applications*, vol. 3, pp. 94-98, 2015.
- [5] J. J. Magee, M. Betke, J. Gips, Matthew R. Scott, and Benjamin N. Waber, "A human-computer interface using symmetry between eyes to detect gaze direction", *IEEE transactions on systems, man, and cybernetics—part a: systems and humans*, vol. 38, no. 6, pp. 1248-1261, 2008.
- [6] H. R. Chennamma and X. Yuan, "A survey on eye-gaze tracking techniques", *Indian Journal of Computer Science and Engineering*, vol. 4, no. 5, pp. 388-393, Nov. 2013.
- [7] B. Champaty, J. Jose, K. Pal and T. A., "Development of EOG base human machine interface control system for motorized wheelchair", *IEEE Annual International Conference in Emerging Research Areas: Magnetics, Machines and Drives (AICERA/iCMMD)*, pp. 1-7, 2014.
- [8] R. F. Ghani and H. S. Hassan, "Human computer interface for wheelchair movement", *Baghdad Science Journal*, vol. 14, pp. 437-447, 2017.
- [9] X. Gao and L. Shi, "The design of robotic wheelchair control system based on hand gesture control for the disabled", *IEEE International Conference on Robotics and Automation Sciences*, pp. 30-34, 2017.
- [10] V. Paul and J. Michael, "Rapid object detection using a boosted cascade of simple features", *Computer Vision and Pattern Recognition, IEEE Computer Society Conference*, vol. 1, pp. 511-518, 2001.
- [11] A. Kasinski and A. Schmidt, "The architecture and performance of the face and eyes detection system based on the Haar cascade classifiers", *Pattern Analysis and Applications*, vol. 13, no. 2, pp. 197-211, 2010.
- [12] B. Javidi, *Image recognition and classification: algorithms, systems, and applications*. CRC press, 2002.
- [13] RASPBERRY PI, 7 March 2018 at 05:00 AM, Available on: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- [14] L298N Motor Driver Board, 7 March 2018 at 05:20 AM, Available on: [https://www.geeetech.com/wiki/index.php/L298N\\_Motor\\_Driver\\_Board](https://www.geeetech.com/wiki/index.php/L298N_Motor_Driver_Board)
- [15] Resolution of an eye, 8 Feb. 2018 at 05:30 PM, Available on: [https://www.wikilectures.eu/w/Resolution\\_of\\_an\\_eye](https://www.wikilectures.eu/w/Resolution_of_an_eye).