ARPN Journal of Engineering and Applied Sciences

DESIGN OF A VISUALLY CONTROLLED ROBOTIC VEHICLE

Divya Mohanakumaran, Suyash Hassija and P. Venugopal
School of Electronics Engineering (SENSE), VIT University, Vellore, Tamil Nadu, India
E-Mail: divya16_m@yahoo.co.in

ABSTRACT

In the design of a visually controlled robotic vehicle, a microcontroller based design is used to send serial data from a computer to a two wheeled robotic vehicle via an Arduino board. This is done to control the vehicle using the color detection technique (blob analysis) via the webcam of a computer on receiving an input from the user. The webcam of the computer detects a particular color of an object placed in front of it. The direction of movement of the robot is decided by the object movements made by the user. This direction is then serially sent via a USB to TTL Converter module from the computer port to the robotic vehicle through the Arduino microcontroller and the motor driver circuit. Hence, the robot can be made to move in all four directions by moving the object of a particular color in front of the webcam in a certain manner. The robotic vehicle also detects obstacle in its path with the help of an ultrasonic sensor put on the back of the vehicle. An ultrasonic sensor transmits ultrasonic waves from its sensor head and again receives the ultrasonic waves reflected from an object in its line of sight. It detects the position of this object by measuring the length of time from the transmission to reception of the sonic wave. Henceforth, when the vehicle is made to move backward, the sensor put on the vehicle detects the object and is triggered to stop when the distance between the object and the vehicle becomes less than 15 cm thereby facilitating safe parking of the vehicle. The visually controlled movements of the robotic vehicle as well as its obstacle detection capabilities make the robotic vehicle worthy of being replaced as a wheelchair for paralytic patients.

Keywords: robotic vehicle, image processing, color detection technique, blob analysis, ultrasonic sensor.

INTRODUCTION

In recent years, image processing and identification techniques have received a lot of attention especially in the moving objects area. The color detection technique is used for people detection and tracking which are important capabilities for applications that desire to achieve a natural human-machine interaction. The use of color greatly reduces the errors of the tracking system. Besides, the low computing time required for the detection and tracking process makes it suitable to be employed in real time applications.

The main objective of the project is to design a robotic vehicle with the help of an Arduino microcontroller and to control it using the color detection technique via the webcam of a computer. The color detection technique (blob analysis) is implemented in the Code::Blocks software after integrating it with the blob library for OpenCV. Detecting the connected components of the binary image is the purpose served by this library. The robot can be made to move to the right, left, forward and backward by moving the object of a particular color (in front of the webcam) to the right, left, up and down, respectively.

SYSTEM OVERVIEW

The visually controlled section consists of webcam and Code: blocks software which does the image processing using OpenCV. The serial communication involves the use of a USB to serial converter module connected between the com port and the Arduino. The Circuit (mechanical section) consists of the Arduino UNO microcontroller, L293D motor driver IC, two DC motors on a two wheeled robot platform powered by a 6 V battery and a 9V battery to the input of the microcontroller.

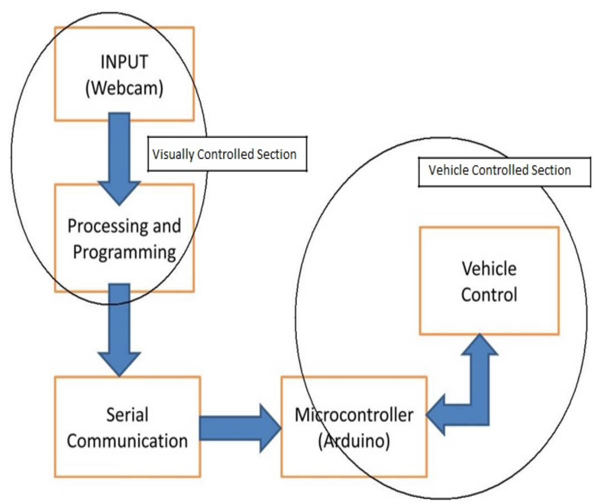

Figure-1. System block diagram.

VISUALLY CONTROLLED SECTION

The purpose is to read an image from the webcam and look for a blob of particular color, say yellow. When yellow color is detected, a frame can be set around the pixels of yellow color and the moments of the image is found out. Whenever there is any motion in the object i.e., there is a change in the image positioning (location of pixels in the frame size set changes) due to the object movement and we get to know whether the direction to be given to the robotic vehicle is left, right, forward or backward by finding out the difference between the two positions.

VOL. 9, NO. 8, AUGUST 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

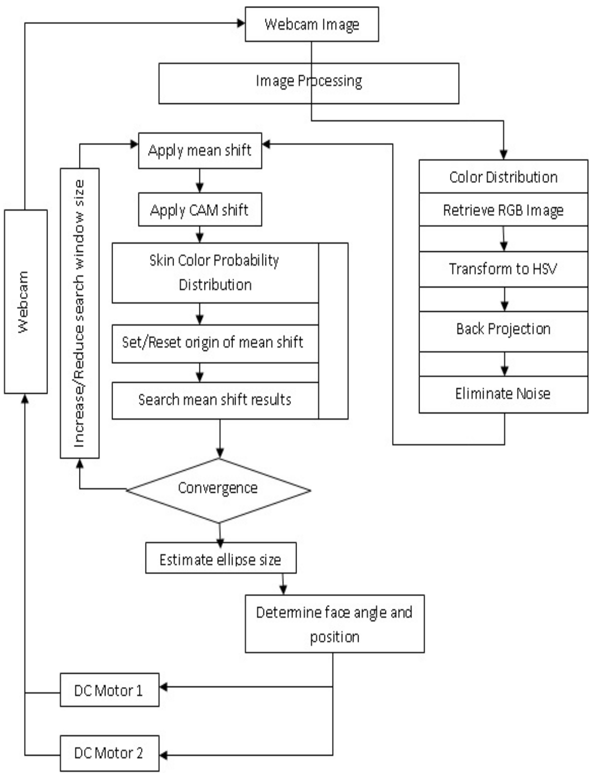

Figure-2. Flow chart of visual platform interface.

SERIAL COMMUNICATION VIA USB TO TTL CONVERTER MODULE

The direction of movement of the robot is sent via this connector from the computer port to the Arduino. The Arduino microcontroller has a code dumped in it which helps in sending the required information to the two dc motors of the robotic vehicle depending on the input received by it from the computer.

VEHICLE CONTROLLED SECTION

A two wheeled robotic platform is built to receive an input from the Arduino. The pins of the microcontroller are configured such that it is responsible for the movement of the wheels of the vehicle according to the instruction obtained from the input (webcam). The vehicle has wheels which are connected to the DC motors. An external battery supply of 6V is given to drive the motors.

ULTRASONIC SENSOR (DISTANCE MEASUREMENT) CIRCUIT

An Ultrasonic sensor provides an easy method of distance measurement. These sensors calculate the time interval between sending the signal and receiving the echo to determine the distance to an object. When the vehicle moves in the backward path and the ultrasonic sensor mounted on the vehicle detects an object, then the vehicle is made to stop when the distance between the sensor and the vehicle becomes less than 15 cm.

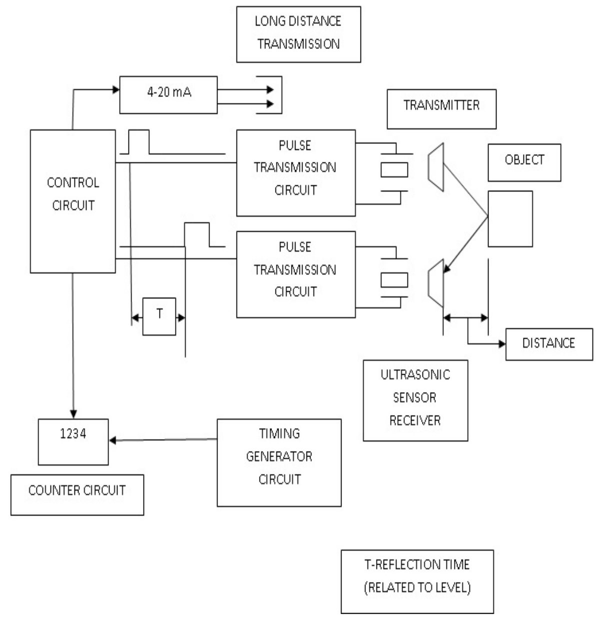

Figure-3. Block diagram of an ultrasonic sensor.

LITERATURE SURVEY

Design of a vision based human machine interface device for a remote control robot is a study in which a computerized vision system is developed to control the orientation of a camera via two DC motors. This vision-based human-machine interface device is able to move in the same direction as the operator head. A sophisticated image-processing technique is the backbone of this device. It locates skin color and separates the face from the complex background. Eventually, the centre of the face is determined as well as the direction of sight. The head position and the judged viewing angle control the motion of the servo motors thereby controlling the orientation of the remote camera. In this study, an integrated surveillance system is presented which consists of a computer, a CCD camera mounted on a xy-platform and a computer interface for communication between the computer and the camera platform. The proposed system directs the camera towards the trouble area using a distant, vision-based, human-machine interface. It also eliminates the problem of blind spots [1]. The authors used the Multi-CamShift method, which is based on the characteristics of color and shape probability distribution to solve the problems of tracking multiple objects. Allen *et al*. reviewed the continuously adaptive mean shift (CamShift) Algorithm and extended its default implementation to allow tracking in an arbitrary number and type of feature spaces. The authors examined the effectiveness of the CamShift algorithm as a general purpose object tracking approach in the case where no assumptions have been made about the target to be tracked. This study uses orientation angle of the head of a human as input and the movement of the machine as the output. When the image is captured by the camera on the human machine interface control platform, it is transmitted to the operator through wireless internet and processed by a computer. The results

ARPN Journal of Engineering and Applied Sciences

of the image processing are then used to control the DC motors on the interface platform through a motion control card. The DC motors controlling the orientation of the camera allow the operator to take pictures at the orientation of his/her choice and transmit them wirelessly through internet [2]. Colored ball position tracking method for goalkeeper humanoid robot soccer is one topic that is often discussed in computer vision, particularly in the development vision for the robot. In a robot soccer game, robots must have the ability to find the position of the ball. The ball usually has a specified color and which is contrast with the color of the field. Robot players must be able to detect the location of the ball that has to be kicked. Goalkeeper robot must be able to detect the exact position of the ball so that it can block the ball from going into the goal. Identifying the location of the colored balls using robot vision is very important along with the mechanical motion of the robot. Detection of colored balls can be done with various methods, such as by detecting a ball through feature detection, or by its color. This paper focused on detecting colored ball object for goalkeeper soccer robot purpose. Image processing hardware in general applies the RGB color model with the consideration of the ease in technical color displaying. Other color model focuses on human's eye perception to the color, like HSV and HSL. HSL color model represents color as three components, Hue (H), Saturation (S) and Lightness (L). Hue is an attribute of human perception and can be described as red, green, blue, purple, and yellow as primary hues or any intermediate combinations of the primary hues. The colorfulness of a color is described by the saturation component. Saturation is thus a measure of colorfulness or whiteness in the color perceived. The Lightness provides a measure of the brightness of colors hence it gives a measure of how much light is reflected from the object or how much light is emitted from a region. Blob detection performed on the images generated from the HSL color filtering. The method used to detect the blob is connected component labeling method. In order to get the largest connected component, each blob is then merged with other blobs that are intersecting each other. This largest blob is considered to represent an object detected by a specific color which in this case is the ball. Then, the center of gravity *CoG* (xc,yc) of the connected component is calculated and is regarded as a point that represents objects position in the coordinate of ball position detection [3]. Detection and tracking of people are important for applications that wish to achieve a natural human-machine interaction. This work is a system that will be able to visually detect and track multiple people using a stereo camera placed at an under the head position. This camera position is appropriate for human and machine applications that require to interact with people or to analyze facial gestures of humans. Initially, the environment (background objects) is modeled using a height map that is later used to easily extract the foreground objects and to search people among them using a face detector. Once a person has been spotted, the system tracks him while it still looks for more people. The tracking is performed using the Kalman filter to estimate the next position of each person in the environment. Nonetheless, when two or more people become close to each other, information about the color of their clothes is also taken into account to track them more efficiently [4]. In real-time visual system for interaction with a humanoid robot, it is currently being used to investigate the ways to program and interact with a humanoid robot. Mimicking or movement limitation of robot and higher forms of learning from demonstrations has been identified as useful tools for programming such robots. The humanoid must be able to detect and perceive human motions to learn from demonstrations and to interact with humans. While off-line processing of visual data is often acceptable for learning from demonstrations, however a real-time system is essential for interaction tasks. Once the motion perception is seen as a continuous process that interacts with the motor system, the required standards of reliability become much more stringent because failure in just one image frame can cause the entire system to break down [5]. The key issue in realizing a real-time motion perception system is to avoid excessive interaction between the different pieces of data in both time and spatial domains. A practical system for perceiving human motion should be one which is able to deal with complex environments and at least moderately changing lighting conditions. Probabilistic approaches are mainly used in this because they prevent excessive data interaction through independency assumptions and also because continuous probabilities associated with image pixels prevent the perceptual algorithms from becoming brittle with respect to the variations in the background and lighting conditions [6-7].

DETECTION OF A YELLOW COLORED OBJECT USING IMAGE PROCESSING

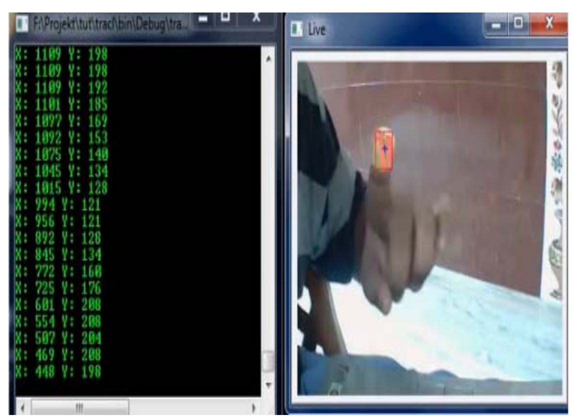

Figure-4. Display of X and Y coordinates on detection and movement of yellow object.

A live feed is taken from the cam and is threshold for yellow color. Then smoothening is done with a median filter and blobs are found out. Then the blobs are rendered so that it can be seen. cvFilterByArea removes any blobs which are too big or too small to be called a blob. Then

VOL. 9, NO. 8, AUGUST 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

their moments are found out, dividing them by their area, the X and Y positions are obtained.

Later, the plane of the live window is divided into four sections as in the coordinate geometry viz. left, right, forward and backward by setting up the values of X and Y coordinates.

When the object is detected in the upper part of the live frame window i.e. positive Y axis, forward is printed. When the yellow blobs are detected on the lower part of the window i.e. negative Y axis, backward is printed and on the left and right parts of the webcam window i.e. positive and negative X axis, left and right are printed respectively as shown below.

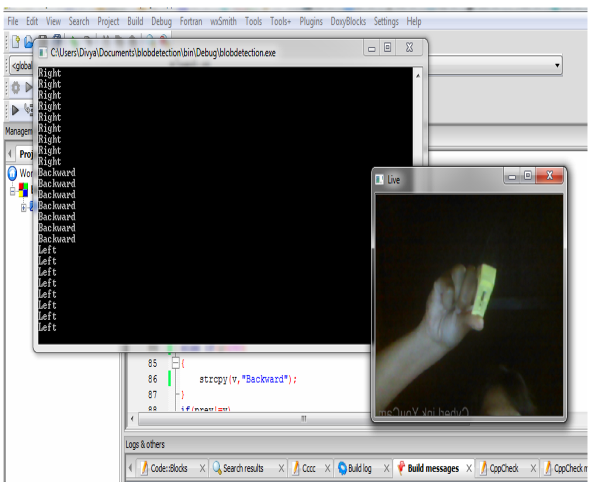

Figure-5. Display of the direction of movement of the robot to be sent to Arduino.

The Arduino code is dumped/uploaded onto the Arduino microcontroller using the Arduino IDE software. The code contains the statements to be performed when a character is serially received from the PC via the USB to serial converter. For instance, when character 'f' is received, i.e. when the vehicle is needed to move forward, both the motors are given the value 'HIGH'. When character 'b' is received, i.e. when the vehicle is needed to move backward, both the motors are given the value 'LOW'. When 'l' is received, i.e. the vehicle has to go to the left, the left motor is given 'HIGH' so that it moves forward and the right motor is given 'LOW' so that it moves backward. Similarly, when 'r' is received, the right motor is given 'HIGH' so that it moves forward and the left motor is given 'LOW' so that it moves backward.

The code is first checked with the mechanical section by sending characters serially through the serial monitor of the Arduino IDE software. The Arduino is then connected to the two wheeled robotic vehicle via the motor driver circuit and the USB to Serial converter module attached to the COM port of the PC. When the vehicle moves in the backward path and the ultrasonic sensor mounted on the vehicle detects an object, then the vehicle is made to stop when the distance between the sensor and the obstacle becomes less than 15cm.

EXPERIMENTAL VERIFICATIONS

Figure-5 shows the output of the color detection technique. The code is written in Code::Blocks software. The code is written to detect an object of yellow color with the help of webcam of the PC. The window frame in which the object is detected is is divided into four quadrants and depending upon the direction of movement of the yellow color object, corresponding instructions are sent to the robotic vehicle.

When the code written in the Code::Blocks sofware is compiled and run, the webcam of the PC opens up and detects the blob of yellow color in the yellow color object as you can see in Figure-5. The directions of movement of the object are forward (positive Y axis), backward (negative Y axis), left (negative X axis) and right (positive X axis).

So when the yellow color object is detected and moved in any of the four directions, the corresponding instruction is first observed in the output window of the Code::blocks sofware and this output is then sent to the microcontroller via the USB to TTL converter module.

When the software gets the command to move the robot forward from the webcam of the PC, then the software sends the letter 'f' to the Arduino microcontroller. Similiarly letter 'b' is sent for backward, letter 'l' is sent for left and letter 'r' is sent for right.

The Arduino Uno microcontroller is connected to the PC and a code is dumped into the microcontroller. The code has instructions to move the motors. The code dumped can control the movement of the motors depending upon the corresponding output of the color detection and direction processing program in the Code::Blocks software.

The robotic vehicle has two DC motors which are connected to the Arduino microcontroller. When the arduino gets the letter 'f' as input from the Code::Blocks software , based on the code dumped, the Arduino sends the instruction to move both the motors in the clockwise direction. Similiarly, when the arduino gets the letter 'b', based on the code dumped, it sends the instruction to move both the motors in the anticlockwise direction. When the Arduino gets the letter 'l', the instruction is given to move the left motor in the clockwise direction and the right motor in the anticlockwise direction and when arduino gets the letter 'r', the instruction is given to move the right motor in the clockwise direction and the left motor in the anticlockwise direction.

SCENARIO 1- FORWARD DIRECTION

When the yellow color object is moved in the postive Y direction with respect to the user, the word "forward" is oberved in the ouput window of the Code::Blocks software which means that the robot has been instructed to move forward. The program in Code::Blocks sends the letter 'f' to the Arduino microcontroller which has been dumped with a code which instructs both the DC motors to move in the clockwise direction. Hence, the robot moves forward. This is shown in the image below.

VOL. 9, NO. 8, AUGUST 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

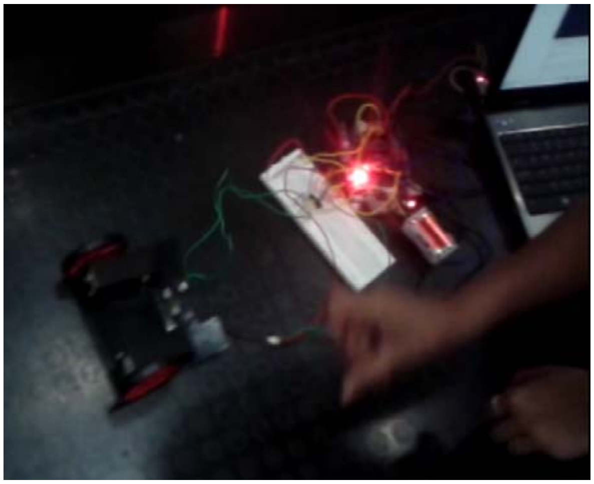

Figure-6. The robot moving in the forward direction.

SCENARIO 2- BACKWARD DIRECTION

When the yellow color object is moved in the negative Y direction with the respect to the user, the word "backward" is oberved in the ouput window of the Code::Blocks software which means that the robot has to be instructed to move in the backward diretion. The codeblocks send the letter 'b' to the Arduino microcontroller which has been dumped with a code which instructs both the DC motors to move in the anticlockwise direction. Hence, the robot moves backward. This is shown in the image below.

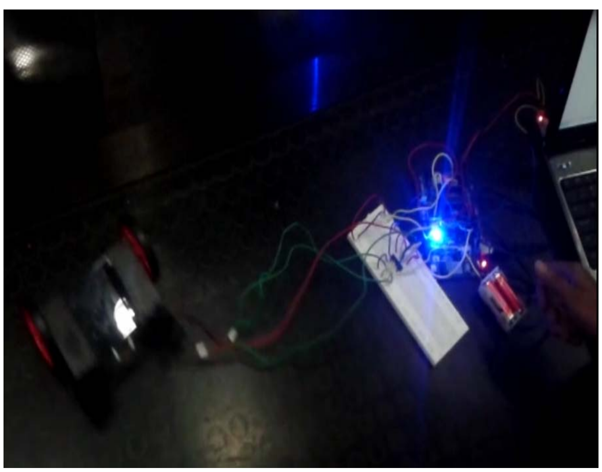

Figure-7. The robot moving in the backward direction.

SCENARIO 3- RIGHT DIRECTION

When the yellow color object is moved in the postive X direction with the respect to the user , the word "right" is oberved in the ouput window of the Code::Blocks software which means that the robot has to be instructed to move in the right direction. The program in Code::Blocks sends the letter 'r' to the Arduino microcontroller which has been dumped with a code which instructs the right DC motor to move in the clockwise direction and the left DC motor to move in the anticlockwise direction. Hence, the robot moves in the right direction. This is shown in the image below.

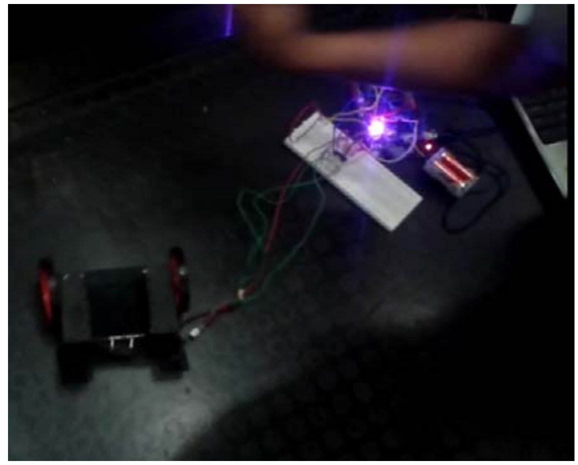

Figure-8. The robot moving in the right direction.

SCENARIO 4- LEFT DIRECTION

When the yellow color object is moved in the negative X direction with the respect to the user, the word "left" is oberved in the ouput window of the Code::Blocks software which means that the robot has to be instructed to move in the left direction. The program in the Code::Blocks sends the letter 'l' to the Arduino microcontroller which has been dumped with a code which instructs the left DC motor to move in the clockwise direction and the right DC motor to move in the anticlockwise direction. Hence, the robot moves left. This is shown in the image below.

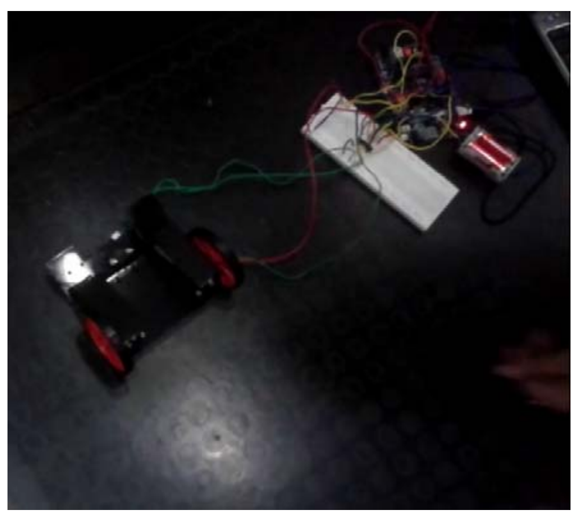

Figure-9. The robot moving in the left direction.

SCENARIO 5- OBSTACLE DETECTION

When the vehicle is made to move in the backward path and the ultrasonic sensor mounted on the vehicle detects an obstacle, it stops and waits for the next input from the user.

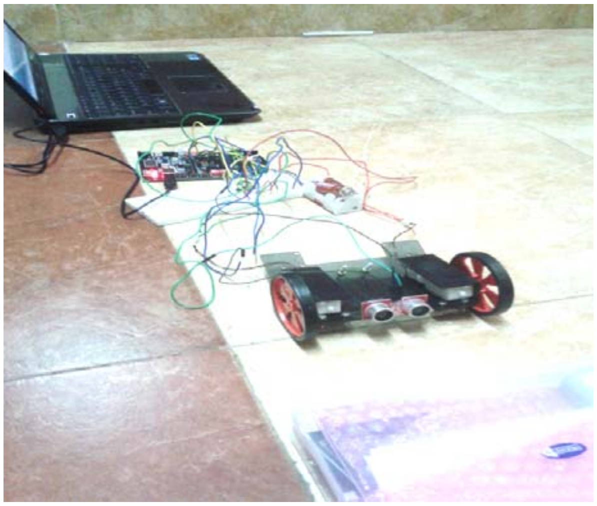

Figure-10. The vehicle stops on detecting an obstacle that is at a distance of less than 15 cm from the sensor.

FUTURE WORK

In the future with the help of new technological advancements, this project can be used to help man in various fields. The color detection technique is used for people detection and tracking which are important capabilities for applications that desire to achieve a natural human-machine interaction. The use of color not only reduces the errors of the tracking system but also lowers the computing time required for the detection of an object and its tracking process. Hence, it becomes very suitable to be used in real time applications. The obstacle detection circuit which is being currently used at a small scale in vehicle for safety parking also holds a lot of use in the future in automated robots and robotic vehicles which can detect obstacles in any direction of its path. Some of the applications are mentioned below.

Real time tracking and mimicking of human movements by a humanoid robot-A humanoid robot system that can capture and mimic the motion of human body parts in real-time. The underlying vision system is able to automatically detect and track human body parts such as hands and faces in images captured .It is based on a probabilistic approach that can detect and track multiple blobs in an image stream. A model is employed to approximate the observed human motion and a Kalman filter is used to estimate its parameters (three-dimensional positions, velocities and accelerations). This enables the system to realistically mimic the perceived motion in real-time.

Colored ball position tracking method for goalkeeper humanoid robot soccer-In a robot soccer game, robots must have ability to find position of the ball. The ball usually has particularly specified color and contrast with the color of the field. Goalkeeper robot must be able to detect the exact position of the ball so that it can block the ball into the goal. Identifying the location of the colored balls using robot vision becomes very important in addition to the mechanical motion of the robot. The next step is detection of ball position for goalkeeper robot soccer to block the ball which can be done using the obstacle detection circuit.

CONCLUSIONS

A visually controlled robotic vehicle is designed by the color detection technique (blob analysis) by taking the input from the webcam of a computer and then moving an object of a particular color, say yellow, in front of it. The input taken is the direction of movement of the robot as desired by the user? This information is then serially sent to an Arduino microcontroller via a USB to TTL converter module to move the robot. The ultrasonic sensor put behind the vehicle helps in safe parking as it stops when it encounters an obstacle which is less than 15 cm from the vehicle. Hence, the robotic vehicle has the capability of moving in all four directions when yellow color blobs are detected and their positions are found out to determine the direction of motion of the robot. The obstacle detection capability as well as the movement of the robot with the color detection technique makes it suitable to be used as a wheelchair for paralytic patients.

REFERENCES

[1] Chen W. 2003. Human face identification and its application in home robot and its application with human being. Master’s thesis, Electrical and Control Engineering, Tatung University.

[2] Rahman Arif and Widodo Nuryono Satya. March 2013. Colored Ball Position Tracking Method for Goalkeeper Humanoid Robot Soccer. Academic Journal, Telkomnika.

[3] Rafael Mu˜noz-Salinas, Eugenio Aguirre, Miguel Garc´ıa-Silvente. 2005. People Detection and Tracking using Stereo Vision and Color. Department of Computer Science and Artificial Intelligence, DECSAI-SI-2005-04.

[4] Aleš Ude, Tomohiro Shibata and Christopher G. Atkeson. 2001. Real-Time Visual System for Interaction with a Humanoid Robot. Robotics and Autonomous Systems. 37: 115-125.

[5] Bradski G. R. 1998. Computer vision face tracking for use in a perceptual user interface. Intel Technol. J, 2nd Quarter. 2(2).

[6] Harshad Sawhney, Sakshi Sinha, Anurag Lohia, Prashant Jalan and Priyanka Harlalka. 2013. Autonomous Rubik's Cube Solver Using Image Processing. International Journal of Engineering Research and Technology (IJERT).

[7] Hewitt R. 2007. Seeing with OpenCV, a Computer Vision Library, SERVO magazine. pp. 62-66.