ISSN:0975-9646

A New Approach to Control a Robot using Android Phone and Colour Detection Technique

Saurav Biswas [1] Umaima Rahman [2] Asoke Nath [3]

[1,2,3]Department of Computer Science,
St. Xavier's College, Kolkata-700016,
West Bengal, India

***Abstract---* Android is a Linux-based, open-source operating system(OS) designed to use on cell phones, e-readers, tablet PCs, and other mobile devices, capable of being used as an alternative for computers having bigger form factors. Android provides a very cheap and easy platform to be the brain for controlling a robot, with arduino being an interface to connect to the H/W and controlling it. In the present paper the authors have designed and implemented the controlling of a robot's movements using distinct colored objects as recognized by the Android Application. This idea may be further extended to control a robot from a remote place also.**

***Keywords---* Android, Color, Coordinate, Application, smartphones, Robot, Hello opencv, Video Stream, Camera Robot, RGB**

I. Introduction

Traditional methods of controlling a robot involve technical expertise and complicated hardware. On the other hand if one can use everyday objects having distinct colors, it would be much simpler and less complicated to control a robot along with decreasing the cost of the hardware. An Android Application recognizes the distinct colored objects and controls the direction of the robot. As a mobile platform, Android has grown in popularity among hardware manufacturers and the general public alike in recent years since it is an open source software stack along with a set of API libraries for writing mobile applications that can shape the look, feel, and function of mobile handsets. Computing power, sensing capabilities and intuitive programming interfaces of Android smart phones afford an inexpensive yet highly capable robotic platform.

A. Constructing the Robot

Preparation of the robot required an Android phone, Arduino BoardUNO (ATMega328), which is connected via Bluetooth Module HC-05 ,Adafruit Motor Shield and a robot chassis. The robot takes advantage of the sensors on the phone (e.g., camera, accelerometers, GPS) via the Arduino.

The various components interact with each other as follows:

1) A smartphone running the Android operating system used as onboard computers and sensing devices.

2) An electronic board (Arduino ADK) used to interact with the motor shield and the Android Phone.

3) A motor shield (Adafruit motor shield v1) to interact with the motors and the Arduino Board

4) A Bluetooth Module used to interact with the Android Phone and the Arduino Board

5) A robot chassis.

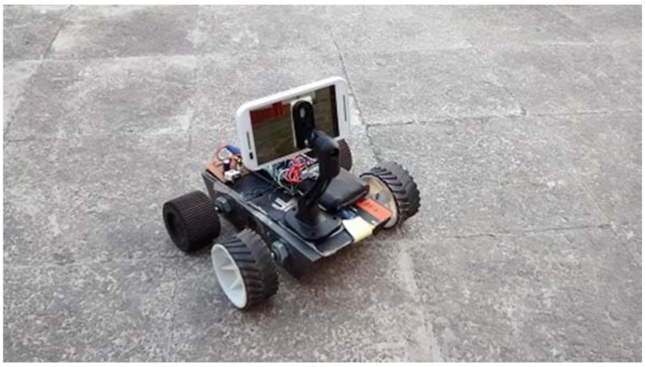


Fig. 1 Picture of the robot that we constructed.

B. Developing the Android Application to detect two different colours for controlling robot's movements.

An Application named "Hello opencv" was developed.

- The Hello opencv Application detects two colors whose RGB Values are fed into the Application.
- The Hello opencv Application with the help of the OpenCV Manger Application converts the given RGB values into HSV(Hue, Saturation and Value) Format.
- OpenCV provides InRange function that can be used to pick out pixels based on their values. Upper and lower threshold values has been provided to this function, it then generates a mask i.e., a binary image where foreground pixels are within the specified range.
- The Open CV then finds all contour points and stores it in an array. Using all the contours , OpenCV forms a rectangle boundary around the selected images.
- Rectangle boundary for the image1 has the coordinates:
  a1(x11,y11),a2(x21,y21),a3(x31,y31),a4(x41,y41)
  Rectangle boundary for the image2 has the coordinates:
  b1(x12,y12),b2(x22,y22),b3(x32,y32),b4(x42,y42)
- The Hello opencv Application then control the robot's direction depending upon any one of the y-coordinate of Image1(say y11) and any one of the y-coordinate of Image2(say y12)
- The bluetooth module receives the String and sends the String to the arduino (microcontroller).
- The arduino (microcontroller) on finally receiving the directions moves the robot along the desired direction.
- The algorithm of controlling the robot's movements is given in Section III. Of this paper.

C. Designing the Graphical User Interface (GUI) of the 'Hello opencv' Application.

For effective execution of the application, a hassle free and an intuitive Graphical User Interface (GUI) is required.

The application connects to the robot through Bluetooth. It then asks for the RGB values of any two colours which is to be focused.

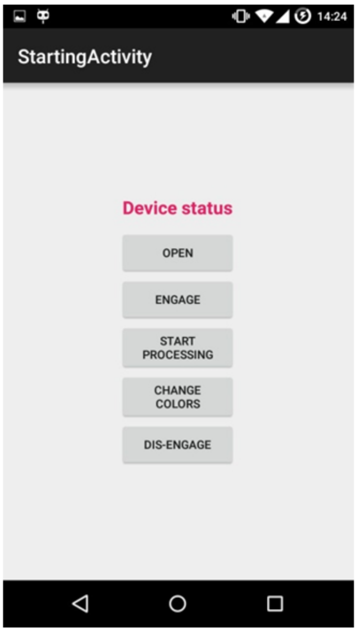


Fig. 2 Screenshot of the 'Hello opencv' Application

Functionality of the buttons present in the 'Hello opencv' Application:

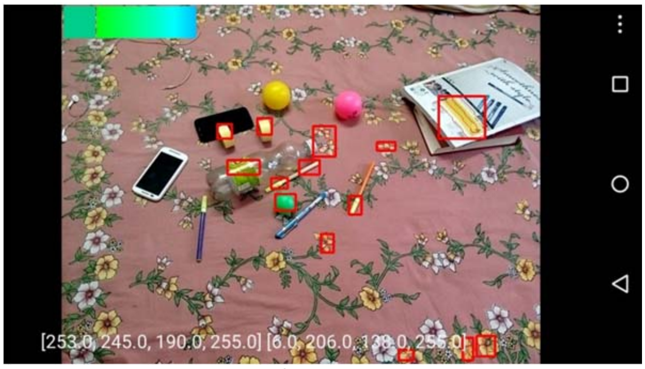
-The "Open" Button switches on the Bluetooth of the phone

- The "Engage" Button connects the phone with the robot through Bluetooth.

- The "Start Processing" Button opens the Camera Application of the phone and focuses on the two colours whose RGB value with which it has been provided.

- The "Change Color" button gives the option to change the RGB values at any time.

-The "Dis-engage" Button disconnects the phone with the robot.



[253.0, 245.0, 190.0, 255.0] [6.0, 206.0, 138.0, 255.0]

Fig. 3 The Application focuses on the given two RGB values once the "Start processing" button is pressed. In this case one of the RGB value is a shade of yellow and the other is a shade of green as marked by red rectangles.

D. Designing and developing Android Applications which can stream real-time video of the robot.

Controlling the robot's movements without seeing it would require an application that can stream a video displaying the robot's location to a screen from where the robot is being controlled.

This requires two Android Smartphones. One is installed on the robot (receiver phone) the other is with the user (controller phone).

The two phones are connected using Wi-Fi connection. The Wi-Fi connection is established using a Wi-Fi router to which the two phones are connected. In absence of a Wi-Fi router, any of the two phones (controller/receiver) is made the Wi-Fi router by enabling the Wi-Fi hotspot option. The other phone is then connected to it.

The receiver phone runs an application to receive the IP address of the device.

The IP address obtained from the receiver phone is inputted in another application which runs on the controller phone along with the password.

On hitting the "Log In" button the connection is established between the two phones.

The receiver phone's camera opens up and streams the video to the controller phone.

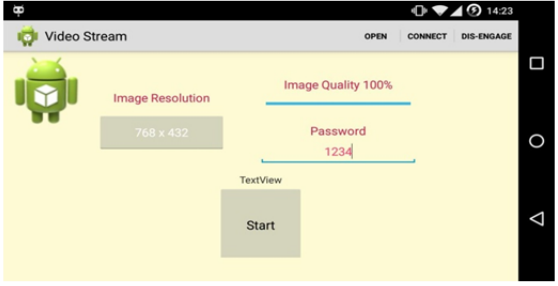


Image Resolution: 768 x 432

Image Quality 100%

Password: 1234

Fig. 4 Screenshot of the 'Video Stream' Application used in the Receiver phone (the phone on the mounted on the robot).

Functionality of the buttons present in the 'Video Stream' Application are as follows:

-The 'Image Resolution' button provides a list of video resolution to choose from.

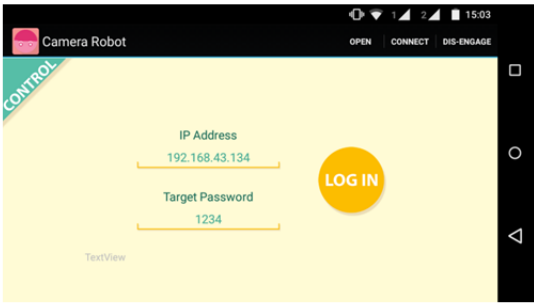
-The 'Start' button opens the camera application of the phone and starts recording.

-A password has to be entered in the text-field for 'Password'

-The "Open" Button switches on the Bluetooth of the phone

- The "Engage" Button connects the phone with the robot through Bluetooth.

-The "Dis-engage" Button disconnects the phone with the robot.



IP Address: 192.168.43.134

Target Password: 1234

Fig. 5 Screenshot of the 'Camera Robot' Application used in the Controller phone(the phone with the Controller).

Functionality of the buttons present in the 'Camera Robot' Application:

- In the text-field provided for the label 'IP Address', the IP Address of the receiver phone's has to be entered.

-In the text-field provided for the label 'Target Password', the same password which was entered in the receiver phone has to be entered.

-The 'LOG IN' button sends the 'Target Password' to the Device whose IP Address has been provided in the text-field for 'IP Address'. If the password matches the receiver phone sends a positive acknowledgement and starts streaming the video. If the password does not match it sends a negative acknowledgement and a message is displayed on the controller phone saying that" connection failed to establish".

-The "Open" Button switches on the Bluetooth of the phone

- The "Engage" Button connects the phone with the robot through Bluetooth.

-The "Dis-engage" Button disconnects the phone with the robot.

II. LITERATURE SURVEY

A. Android SDK

The true appeal of Android as a development environment lies in the APIs it provides.

As an application-neutral platform, Android gives the opportunity to create applications that are as much a part of the phone as anything provided out of the box. The following list highlights some of the most noteworthy Android features:

- No licensing, distribution, or development fees
- Wi-Fi hardware access
- Comprehensive APIs for location-based services such as GPS
- APIs for accelerometer and compass hardware
- IPC message passing
- Shared data stores
- An integrated open source Web Kit-based browser
- Full support for applications that integrate Map controls as part of their user interface
- An application framework that encourages reuse of application components and the replacement of native applications.

Using Eclipse with the ADT plug-in for Android development offers some significant advantages.

Eclipse is an open source IDE (integrated development environment) particularly popular for Java development. It's available to download for each of the development platforms supported by Android (Windows, Mac OS, and Linux) from the Eclipse foundation homepage:
www.eclipse.org/downloads/

B. Arduino IDE

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on processing and other open-source software.

C. OpenCV

Open Source Computer Vision is a library of programming functions mainly aimed at real-time computer vision, developed by Intel Russia research center in Nizhny Novgorod, and now supported by Willow Garage and Itseez. It is free for use under the open-source BSD license. The library is cross-platform. It focuses mainly on real-time image processing.

III. ALGORITHM

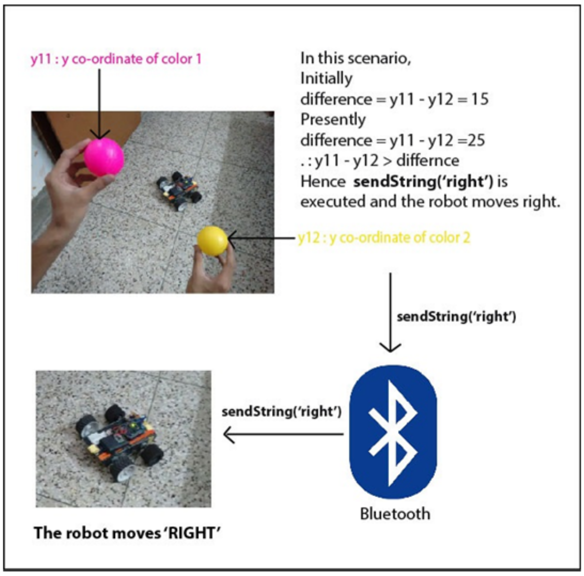
A. Algorithm for controlling the movements of the robot by detecting two colours using the 'Hello opencv' Application.

```
// contours1.size() is the area of the first color.
//contour2.size() is the area of the second color.

if    (contours1.size()    ==    0    &&
contours2.size() == 0)
       sendString('stop');

       // Stops the robot
// if both the colors are out of focus then the robot is made
to stop.

else   if   (contours1.size()   ==   0   ||
contours2.size() == 0)
       sendString('reverse');

       // Moves the robot in the Reverse direction
// if anyone color is out of focus then the robot is made to
traverse in the reverse direction.

Difference = difference between y11 and
y12;
//initial difference between any of the y coordinates of the
first color and the y co-ordinate of the second color

if (y11 <  y12)
{
if (y12 -  y11  > difference )
       sendString('left');
       // Moves the robot in the Left direction
else
       sendString('forward');
       // Moves the robot in the Forward direction
}
else
{
if (y11 -  y12  > difference )
       sendString('right');
       // Moves the robot in the Right direction
else
       sendString('forward');
       // Moves the robot in the Forward direction
}
```

B. Algorithm for moving the robot depending upon the string received.

// There are four motors in the robot:
//motor1, motor2, motor3, motor4
//Depending on the string , the motors are made to move
//forward, backward or is released.

```
if  (string = ‘left’)
{
 motor1.run(FORWARD);
 motor2.run(BACKWARD);
 motor3.run(BACKWARD);
 motor4.run(BACKWARD);
}
 if  (string = ‘right’)
{
 motor1.run(BACKWARD);
 motor2.run(FORWARD);
 motor3.run(FORWARD);
 motor4.run(FORWARD);
}
 if  (string = ‘forward’)
{
 motor1.run(FORWARD);
 motor2.run(BACKWARD);
 motor3.run(FORWARD);   motor4.run(FORWARD);
}
 if  (string = ‘reverse’)
{
 motor1.run(BACKWARD);
 motor2.run(FORWARD);
 motor3.run(BACKWARD);
 motor4.run(BACKWARD)
}
 if  (string = ‘stop’)
{
 motor1.run(RELEASE);
 motor2.run(RELEASE);
 motor3.run(RELEASE);
 motor4.run(RELEASE);
}
```



In this scenario,
Initially
difference = y11 - y12 = 15
Presently
difference = y11 - y12 =25
.: y11 - y12 > differnce
Hence **sendString('right')** is
executed and the robot moves right.

Fig. 6 Schematic describing the execution of the project

IV. Experimental Results

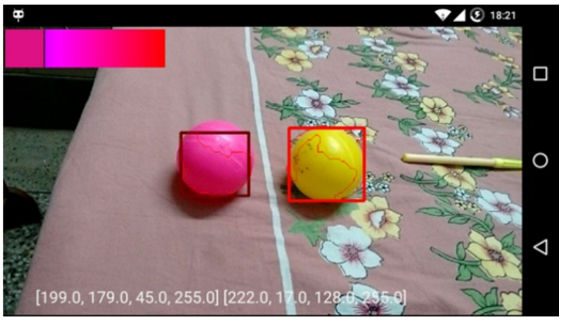

[199.0, 179.0, 45.0, 255.0] [222.0, 17.0, 128.0, 255.0]

Fig. 7 Hello opencv Application detects two different colored objects. It finds the Contour Points. Then makes a rectangle around the object using the contour points of the object.

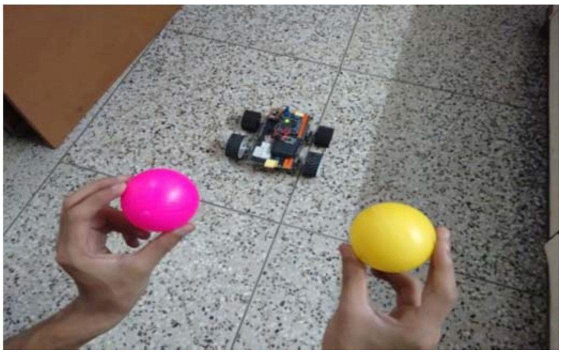


Fig. 8 The y co-ordinate of object1 is equal to the y co-ordinate of object2. Hence, the robot moves FORWARD

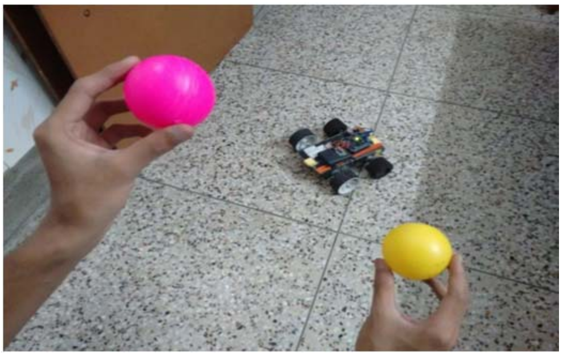


Fig. 9 The y co-ordinate of object1 is less than the y co-ordinate of object2. Hence, the robot moves LEFT

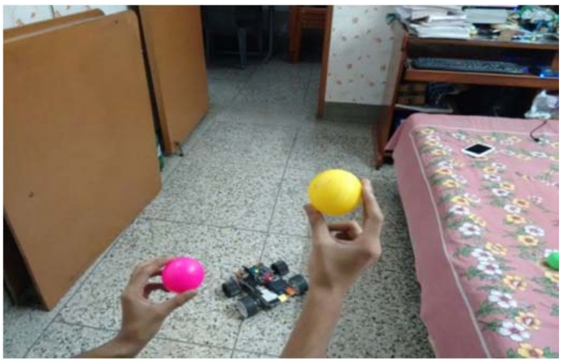


Fig. 10 The y co-ordinate of object1 is greater than the y co-ordinate of object2. Hence, the robot moves RIGHT

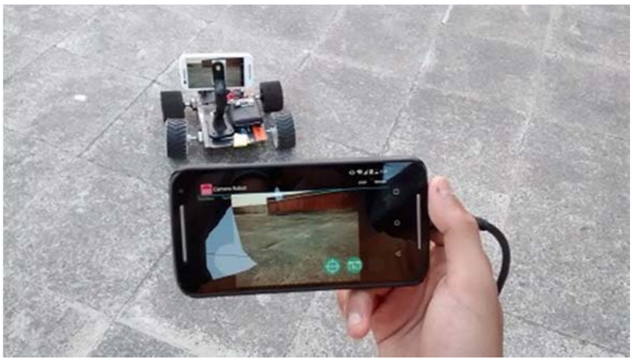


Fig. 11 Video captured by the phone mounted on the robot is being displayed in the Application in the Controller Phone.

V. Conclusion And Future Scope

The experimental results show that the robot moves in accordance with the motion of the y-axis of the distinct coloured objects (in this case two balls of different RGB values).

The robot's movements can be controlled even if it is out of sight by using the' Video Stream' and the 'Camera Robot' Application. These Applications help to stream real time videos and make the control easier.

There can be many applications of this project with further enhancements according to one's requirements.

As Android is an open source OS developed by GOOGLE suggests that the OS can be tweaked and modified when and where required provided with necessary skills.

Android has been in many fields from the previous year and not just constraining itself to the smartphone domain.

- Cars(Android-Auto).
- Watches(Android-Wear).
- Home Appliances(Currently Room Heater and Home Appliances).

The authors have started work to control robot from a remote area through internet. This may help to control various equipments from a remote place.

References

[1] Android apk tool: A tool for reengineering Android apk files. code.google.com/p/android-apktool/.

[2] Google Android SDK, http://developer.android.com/sdk/index.html

[3] In U.S. market, new smartphone buyers increasingly embracing Android. blog.nielsen.com/nielsenwire/online mobile/, sep 2011.

[4] ANDROID OPEN SOURCE PROJECT. Platform versions. developer.android.com/resources/dashboard/ platform-versions.html.

[5] CHIN, E., FELT, A. P., GREENWOOD, K., AND WAGNER, D. *Analyzing inter-application communication in Android*. In Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (New York, NY, USA, 2011), Mo- biSys '11, ACM, pp. 239–252.

[6] Quality Software, 2001. Proceedings.Second Asia-Pacific Conference, http://ieeexplore.ieee.org/