Android Application Based Remote Monitoring System Using Raspberry Pi3

S.Srividhya[1],G.Dinesh Kumar[2], J.Manivannan[3], V.A.Mohamed Wasif Rihfath[4], K.Ragunathan[5]

[1]Assistant Professor , [2]Project Scholar

[1,2]Department of Computer Science and Engineering,

Cheran College of Engineering, Karur, India

[1]*srividhya.apcse@gmail.com ,* [2]*dynamicdinesh5@gmail.com,* [3]*manivannanjeevanantham007@gmail.com,*

[4]*mohamedwasif22@gmail.com,* [5]*ragubilla011@gmail.com*

Abstract – This paper described about a robot used for remote monitoring system. The robotic car is designed Wireless controlled. It can be moved in all directions and gets a live stream on android platform using Raspberry Pi with tilt mechanism. The android application allows us to control the RPI camera with all directions using Raspberry pi3. These received signals are processed by the commands and the robot is directed with user directive. Existing system has CCTV camera which is fixed on and needs more in numbers to monitor. But robot is movable and it is very easy to cover all the directions. So the issues of existing system is solved. The high streaming video will be obtained in the proposed system.

Keywords – Raspberry Pi, Android, smartphone

1. INTRODUCTION

As we known, mobile platform has become a limb which all humanity from young to old cannot give up. Therefore, it is the case that an application made on this platform will be more interesting,

enjoyable and more useful for them. The development of smartphones o'ers a preliminary opportunity for low cost robots. Some of them are the WI-FI and camera systems which will be used in the project. Besides, it has a stronger CPU. This means that it expresses speed in terms of data communication. Robots are being used increasingly in our everyday life and also in industrial application. These robots which are enjoyable to use in terms of fun for humans especially young generation are controlled wirelessly using WI-FI. So this feature makes them to use robots freely and uninterrupted. As hardware, the robot body is build mechanically and electrical components were also used to build the it, Mostly the robots are controlled by wired these wired robots have some space limitation. So to avoid the limitation, the robotic control is made wireless that is; it is controlled by WI-FI. Wirelessly also means using Bluetooth but the advancement used here is the WI-FI which is most widely used nowadays. Using L298N motor driver boards, the Robot is controlled through the GPIO pins. This robot can be controlled through a smartphone and Raspberry pi communicating between them. So an android application is developed in the Android platform. Here Android application being the command center of the robot as it commands the car to move in certain directions through Android JAVA language. IN this project, Android supported device is used as a robot brain. At this point, it is possible to talk about one and only communication channel. It is between the WI-FI which Raspberry pi and smartphone connects as common and the smartphone. When we consider these operations, the first aim is to be able to control the Robot by using WI-FI, the second one is to be able to get a live video stream from the camera by using Raspberry pi internet address.

2. THE PROCESS OF CAR

Under the process of car, it is possible to mention two parts as hardware and software. In the hardware part we will talk about the requirements for building the robots and the building process. In the software part, we will talk about separately our main communication items Android and Python which send the signals to each other.

A. HARDWARE PARTS
1) *REQUIREMENTS*: We have tried to show the modules for motors which are necessary for their working mechanism and their numbers within the scope of this project on the table 1 below;

TABLE 1:REQUIREMENTS

Number	Needed
1	Raspberry pi 3
1	L298N Motor Driver
1	USB Camera
4	DC Motors

a)Raspberry pi: A Raspberry pi which is shown Figure 1 is a credit card-sized computer originally designed for education. Thanks to its small size and accessible price, it was quickly adopted by tinkerers, makers, and electronics enthusiasts for the require more than a basic micro-controller (such as Arduino devices). [1]

Figure.1 Raspberry Pi 3

The Raspberry is slower than a modern laptop or desktop but is still a complete linux computer and can provide all the expected abilities that implies, at a low-power consumption level. It is also known as microcomputer. It is central unit of the system. [1]. Some technical specifications about raspberry pi 3 are given below Table II;

TABLE II: TECHNICAL SPECIFICATIONS

Processor	4×ARM Cortex-A53, 1.2GHz
Memory (RAM)	1GB LPDDR2 (900 MHz)
Internet Connection	10/100 Ethernet, 2.4GHz 802.11 n wireless
Memory	Micro SD
GPIO	40-pin header, populated
Other Ports	HDMI, 4× USB 2.0, Ethernet, Camera

Raspi pins which are shown figure 11; a physical interface between the pi and the outside world. Seventeen of the 26 pins are GPIO pins; the others are power or ground pins. [2]
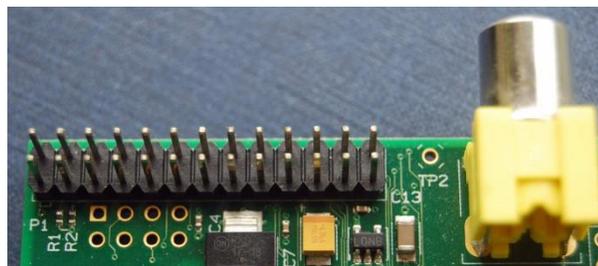

Figure.2 Raspberry Pi Pins

International Journal of Computer Science and Engineering Communications
Volume.6, Issue.1 (2018): Page.1802-1812
www.ijcsec.com

b) L298N Motor Driver: This allows you to control the speed and direction of two DC motors. The L298N H-bridge module which is shown in Figure 3 can be used with motors that have a voltage of between 5 and 35V DC. [3]

There is also an onboard 5V regulator, so if your supply voltage is up to 12V you can also source 5V from the board.[3]

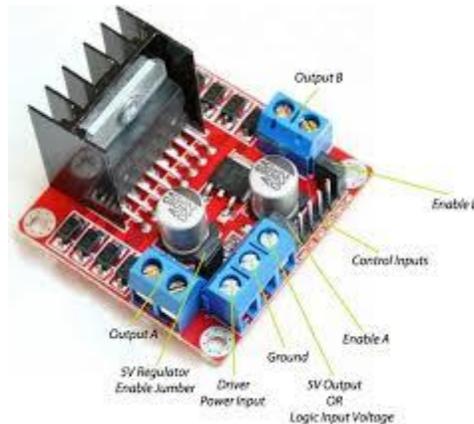The input, output pins on L298N are shown step by step below. [3]

Figure.3 L298N Motor Driver Pin

Pins:

1 DC motor 1 "+" or stepper motor A+

2 Dc motor 1 "-" or stepper motor A-

3 12V jumper – remove this if using a supply voltage greater than 12V DC. This enables power to the onboard

4 Connect your motor supply voltage here, maximum of 35V DC. Remove 12V jumper if>12V DC

5 GND

6 5V output if 12V jumper in place, powering your Arduino (etc)

7 DC motor 1 enable jumper. Leave this in place when motor speed control.

8 IN1

9 IN2

International Journal of Computer Science and Engineering Communications
Volume.6, Issue.1 (2018): Page.1802-1812
www.ijcsec.com

10 IN3

11 IN4

12 DC motor 2 enable jumper. Leave this in place motor speed control.

13 DC motor 2 "+" or stepper motor B+

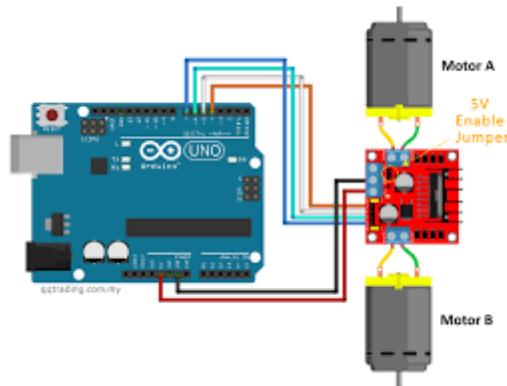14 Dc motor 2 "-" or stepper motor B

Figure.4. L298N – DC Motor Connection

Following Figure 4 shows us how to connect dc motors to L298N Motor Driver.

    Speed control function are connected with the GPIO 17. GPIO 23 interfaces. Direction control function are connected with the GPIO 22, GPIO 24 interfaces. This is shown as the following table III:

Function	Channel A Pin	Channel B Pin
Direction	GPIO 22	GPIO 24
PWM	GPIO 17	GPIO 23

 c) DC Motors: The most preferred type of motor in robotics is DC motors shown in Figure 5. They are cheap, small, elective and also very various in terms of size, shape will explain dc motors in terms of direction, speed, voltage and current.

International Journal of Computer Science and Engineering Communications
Volume.6, Issue.1 (2018): Page.1802-1812
www.ijcsec.com

Figure.5. DC Motor

Direction: When a power supply is connected to the DC motors, the direction of rotation of the DC motor depends on the direction of the current. When the direction of the current is reversed. The direction of rotation of the Dc motor is reversed.

Speed: The speed of motor is measured by the number of revolutions completed in a minute. [4] The speed of the motor depends on then voltage and load.

Voltage: Small Dc motors can be found with voltage values ranging from 1.5 V to 48 V. [4] While using Dc motors is robots and other systems, this voltage value is important because it determines the maximum operating voltage to be applied to the DC motor.

Current: When a DC motor is operated at the specified voltage, the current of the DC motor depends on the load. If the load increases, the current by the DC motor increases. The DC motor should not be overloaded to exceed the maximum current limit. In such a case, the Dc motor is short-circuited and the applied power turns into heat. This may cause the DC motor to burn for long periods of time. Generally, the applied current range of DC motors can be up to 50Ma and above 2A.[4]

d) USB camera: It will be possible to transfer live video to the phone with the USB cam shown in figure 6 when the necessary operation is done thanks to the usb connection which is connected to Raspberry pi WI-FI.

Figure.6. USB Camera

2) Car Building Process: On the car building side, the first step shown as figure 7 below is to mount the dc motors which are wired, to the chassis.

The other steps shown as Figure 8 are to mount the dc motors to the L298N Motor Driver, Raspberry pi to the chassis and USB with wheels to the chassis.
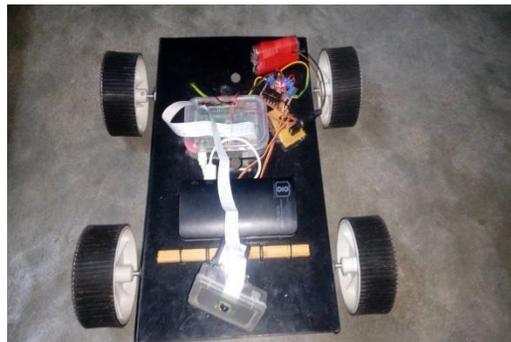

Figure 7. L298N, Raspberry Pi, USB Cam

B. *SOFTWARE PARTS*

1) *ANDROID APPLICATION*: In Android Application, it is possible to mention two parts as Android Code Flows and User Interface. In the android Code Flows part we will talk about what does each JAVA class do in Android Application. In the user interface part, we will examine how the application looks visually.

Android code Flows: This field contains everything from the birth to the end of the Android App. The design consists of 3 separate layouts.

. activity_intro.xml

.activity _robot_control.xml

. activity_fragment_buttons.xml

As we know, each screen designed in layout includes its own software. So it is necessary to open the class at least as much as it is. The first of these is the Splash Activity class, which corresponds to the activity_intro.xml layout. When we open the app, we encounter this activity first time, and it goes to the main layout with the transition time of 3 seconds. Seconds we have an e num class named Instruction State that specifies the command state. We can talk about two things in this class. These values are delivered to server. After that it waits to map.

Moving ("+"): To move

Moving ("-"): To Stop

Third, we have e num class Robot Instruction which maps server robot commands to e num. These values are used to indicate in the python file on the server side.

Forward ("FF", "Forward")

Back ("BB", "Back")

Left ("LL", "Left")

Right ("RR" ,"Right")

Speed ("SS" , "Speed", Button Type. Value)

Fourth we've thread class Client Socket Thread, The class that provides the actual communication with the server. It provides the actual connection using threads and sockets with the set host and open port. In doing so, it uses the Print Writer class, which is part of android to send data to the external source. This command is used to send the external source. This command is used to send Fifth, In Android interface we have two separate button type, first one is touch button, second one is value. Touch button is used to control car, the other is to control speed of robot with Seek bar .Therefore this e num class named Button Type is used to indicate them.

Sixth, this class named Button Actuator keeps a list of buttons and dispatches command to them on up-down events. Firstly it creates the map of buttons to commands. Thanks to key value which is created from Hash map function, each button behaviors are observed. According to designed motion, if user finger has been lifted o' (up), it sends command to stop movement. else if button has been pressed and held down(down),it sends command to turn on movement. Seventh this class named Button Instruction holds robot command for the button plus its state. And it decides which button does which activity such that touch button controls robot and value button adjusts the speed.

The eighth of these is the MjpegStream. This class be-gins by describing he Mjpeg input stream, which is not natively supported by android. In doing so, it reads each next frame coming from the input stream. URI (Uniform Resource Identifier - http://192.168.43.209;8081 (This url which comes from phone WI-FI is adjusted as static in Raspberry Pi)) is sent to the client and an http entity is created with the http response incoming from the client. The input stream in this http entity is sent into the Buffered Input Stream which is refilled many bytes at a time and then to the Data Input Stream. At the end of this, the URI will complete the image acquisition in the background.[6]

Ninth, The class Control Buttons Fragment in which the motion buttons Fragment in which the motion buttons and Seek Bar are created as a view within a fragment. This class Introduces the buttons and they belong which feature (forward, backward, right, left) before switching to the main layout. Also it corresponds withactivity_fragment_buttons.xml.

The tenth and the last class, Robot Control Activity class, is the class that the user faces on the project on the android phone. When we look on the screen, it is possible to see the 4buttons which are necessary for controlling the robot, I Seeker, live camera video stream and the refresh link button as an extra. First, We interact with the constructor we defined in the video stream in the MjpegStream Class and the Surface view.

We created in the Activity_robot_control.xml class. In the Raspberry Pi Server, we receive live video with IP which we defined as static. So we defined this IP in its host. Finally, we call the button actuator class to send the signal to the server for control commands so we have to control the speed and set the speed. User interface: The application starts with the intro picture. After that you will see the last intent to control the car. In the last intent, besides the view of the camera, it be possible to control the car with the buttons. Hint: Left side buttons express left-right direction of car and right side buttons express forward-backward direction of car.

2) PYTHON SCRIPT: We use the Python Script server section to control the DC motors connected to the Raspberry Pi pins via L298N with the necessary motor control commands to process the data from Android. It is possible to mention two python files here.

. motorPWM.py

. motorServer.py

If we are going to explain the motorpwm.py file first, here we use the library called GPIO which is the port used for communication and control of electronic devices on the raspberry pi card, General Purpose Input/ Output (GPIO).Thanks to this library it is possible to control the pins connected to raspberry pi. In this class we will also able to change the speed of the robot by changing the speed of the robot by changing the speed from android with the PWM (Pulse-Width Modulation) feature. [7]

Second, the robot server.py file is the actual server file from which dc motors prompts are called and which Android tries to communicate with. The file is connected to the host and port specified by the blind () command. Connection requests from Android are listened to. Our main goal is to reach the FF, BB, LL, RR, SS movement commands we have defined on Android side. When doing this, it also checks whether there are enough number of commands come from Android. If the number of commands is equal to "+" (Moving ("+") defined in the android instruction state class), run the commands. If the command number is equal to "-"means that it does not run commands. [8]

III. EVALUATION

The users that will be play with the system could be child or adults, they just have to know how to use a smart phone, launch an application and push some buttons. There is no really limitation about the age while they know how to use the phone. As is known, this project has strengths and weakness as it is in every project. From the strength side we can easily say that the project works, we have made what we wanted. In terms of getting live view the project has good and speed video stream. With the video streaming width designed to fill the user's phone screen, the user will be able to control the car comfortably. We need communications to activate these clever things. In this process we have used only WI-FI technologies that is supported by smart phones and Raspberry Pi. Thanks to this technology It connects to Raspberry Pi without difficulty and never breaks unless we are away from a certain distance. From the weakness side, when a connection is made with the phone (personal access point wirelessly) in places where there is no WI-FI, if the distance between the computer and the Raspberry Pi phone are more than 10-15 meters, the connection may weaken and break.

IV. CONCLUSION AND FURTHER WORKS

Everything that is smart makes our life easier and more meaningful. Thanks to Raspberry Pi Technology, there are now so many systems that make people's lives easier and almost inventive, which is impossible to catch up with. In this project, We have accomplished to design our own car using Raspberry Pi and to use it in mobile environment and Future works for this at process will focus on the evaluation and application of the improved techniques. Our goal here is to contribute to the development of the robot by showing up everything that smart cards can do and see how far we can go. So the additional studies projected for the future in our thinking are in the following order: Add sounds to warn the user on some events such as "out of range", "video lost", etc.Improve the application with new modules such as Proximity detection using sensor. Some sensors of the Arduino can be added to bring joy to the car. (honk-buzzer, distance-ultrasonic, headlights-lamp sensor etc.,).Arm control can be added. Finally, We have learned many things in several parts, hard-ware or software, even project management, this project helped us, to better understand the communication with hardware and software. We are used to develop software and we are not used to deal with hardware, it's also really complicated many information and documentation to read, to be aware of. You cannot really break software, it doesn't happen often, but it's really easy to break hardware for instance, We have discovered that.

REFERENCES

[1] What is a Raspberry Pi,   https://opensource.com/resources/what-raspberry-Pi, June 2017

[2] An Introduction to GPIO and Physical Computing on the Raspberry Pi, https://www.raspberrypi.org/documentation/usage/gpio/, June 2017

[3] Control DC and Stepper Motors with L298N Dual Motor Controller http://www.instructables.com/id/Control-DC -and-stepper-motors-with-L298N-Dual-Moto/, June 2017

[4]DC Motorlar, http://www.robotiksistem.com/dc_motor_ozellikleri.html,january 2017

[5]Android Java Client Program using Separate Thread for Socket IO, http://stackoverflow.com/questions/22110205/android-java-client-program-using-seperate-thread-for-socket-io-cant-find-a-wa,june 2017

[6] Android and MJPEG, https://stackoverflow.com/questions/3205191/android-and-mjpeg,February 2017.