

PATROLLING ROBOT FOR LATE NIGHT CRIME MONITORING AND LIVE STREAMING

Submitted in partial fulfilment of the requirements for the award of
Bachelor of Engineering Degree in Computer Science and Engineering

By

MONISHA K (Reg. No. 38290046)

JESMI SHEHA C (Reg. No. 38290030)



**DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING SCHOOL OF COMPUTING**

**SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU**

MARCH 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

(Established under Section 3 of UGC Act, 1956) JEPPIAAR NAGAR,
RAJIV GANDHI SALAI, CHENNAI- 600119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **MONISHA K (Reg. No. 38290046)** and **JESMI SHEHA C (Reg. No. 38290030)** who carried out the project entitled "**PATROLLING ROBOT FOR LATE NIGHT CRIME MONITORING AND LIVE STREAMING**" under our supervision from November 2020 to March 2021.

Internal Guide

Mr.R.Vignesh,M.E.,(Ph.D).,

Head of the Department

Dr.S.Vigneshwari,M.E,Ph.D and Dr.L.Lakshmanan,M.E,Ph.D

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I ,**MONISHA. K(reg. no. 38290046)** hereby declare that the project report entitled “**patrolling robot for late night crime monitoring and live streaming**” done by me under the guidance of **Mr.R.VIGNESH, M.E.,(PH.D)**, is submitted in partial fulfillment of the requirements for the award of bachelor of engineering degree in computer science and engineering.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. SASIKALA, M.E., Ph.D.**, Dean, School of Computing and **Dr. S.VIGNESHWARI, M.E., Ph.D.**, and **Dr. L. LAKSHMANAN, M.E., Ph.D.**, Head of the Department, Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Mr.R.Vignesh, M.E., (Ph.D)** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Computer Science and Engineering who were helpful in many ways for the Completion of the project.

ABSTRACT

Robberies, burglaries and thefts continue to plague the length and breadth of this country. As per the National Crime Records Bureau (NCRB), 2,44,119 cases of robbery, theft, burglary, dacoity, among others, took place in residential premises in 2017. This was a jump of over 10% from 2016 when the number of such cases stood at 2,20,854. The financial loss due to these thefts and burglaries are staggering. In 2017, value of property stolen from residential premises was in excess of Rs. 2065 crores, a 40% jump from Rs. 1,475 crores stolen the previous year. Indeed, home safety is a serious concern for the residents as well as the police and the authorities. To overcome this problem we have developed a project that gives a technological solution using a robot. In this project we develop a robot with a hardware module connected with a camera and sound sensors embedded to it. A mobile application is developed using react native. The robot is given a specified distance to patrol around repeatedly, in case an abnormal noise is detected by the sound sensor the robot immediately alerts the nearby police personnel with an alert and also provides a live streaming through the mobile application.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO.
	ABSTRACT	i
	LIST OF FIGURES	ii
	LIST OF SYMBOLS AND ABBREVIATION	iv
1	INTRODUCTION	1
	1.1 GENERAL	1
	1.2 TECHNOLOGY USED	4
	1.2.1 Robotics	5
	1.2.2 OpenCV	8
	1.2.3 Embedded systems	
	1.2.4 MQTT	
	1.2.5 React native app development	
	1.3 Objective	12
2	2.1 INTRODUCTION	13
	2.2 LITERATURE SURVEY	13
	2.2.1 Haibing Wu , Bo Tao , Member, IEEE, Zeyu Gong, Member, IEEE, Zhouping Yin, Member, IEEE, and Han Ding, Senior Member, IEEE,” A Standalone RFID-Based Mobile Robot Navigation Method Using Single Passive Tag”	13
	2.2.2 Asif Khan, Bernhard Rinner, Senior Member, IEEE, and Andrea Cavallaro,” Cooperative Robots to ObserveMoving Targets: Review”	14

2.2.3 Shuai Li, Ruofan Kong, and Yi Guo, Senior Member, IEEE,” Cooperative Distributed Source Seeking by Multiple Robots: Algorithms and Experiments” 15

2.2.4 Olzhas Adiyatov , Student Member, IEEE, Bexultan Rakhim, Altay Zhakatayev , Member, IEEE, and Huseyin Atakan Varol , Senior Member, IEEE,” Sensor Reduction of Variable Stiffness Actuated Robots Using Moving Horizon Estimation” 16

2.2.5 J. García, A. Bohórquez, and A. Valero,” Suspension Effect in Tip-Over Stability and Steerability of Robots Moving on Sloping Terrains” 17

2.2.6 Milan Lukic, Ahmed Barnawi, Member, IEEE, and Ivan Stojmenovic, Fellow, IEEE,” Robot Coordination for Energy-Balanced Matching and Sequence Dispatch of Robots to Events” 16.7

8

2.2.7 Jing Li; Qiang Huang; Zhangguo Yu; Xuechao Chen; Weimin Zhang; Huaxin Liu; Junyao Gao; Yingxian Duo,” Integral Acceleration Generation for Slip Avoidance in a Planar Humanoid Robot” 19

2.2.8 Guillaume Doisy; Joachim Meyer; Yael Edan,” The Impact of Human–Robot Interface Design on the Use of a Learning Robot System ” 20

2.2.9 Feng Gu; Yuqing He; Jianda Han,” Active Persistent Localization of a Three-Dimensional Moving Target Under Set-Membership Uncertainty Description Through Cooperation of Multiple Mobile Robots” 21

2.2.10 C. Serkan Karagöz; H. Işıl Bozma; Daniel E. 22

	Koditschek,” Coordinated Navigation of Multiple Independent Disk-Shaped Robots”	
	2.2.11 Guanglong Du; Mingxuan Chen; Caibing Liu; Bo Zhang; Ping Zhang,” Online Robot Teaching With Natural Human–Robot Interaction”	23
	2.2.12 P.H.L. Silva; A.M.F. Burlamaqui,” System for Creation, Programming and Availability of Distributed Robot Teams using Collective Knowledge ”	24
	2.2.13 F.A.G. Becerra; A. B. Ortega; C. D. G. Beltrán; C. G. Valdivia; R.O.D. Arcega,” Design and control of a new parallel robot for the rehabilitation of the hip-knee”	25
	2.2.14 Takayuki Kanda, Member, IEEE, Dylan F. Glas, Masahiro Shiomi, and Norihiro Hagita, Senior Member, IEEE,” Abstracting People’s Trajectories for Social Robots to Proactively Approach Customers”	26
	2.2.15 Matthew Field; David Stirling; Zengxi Pan; <u>Fazel Naghdy</u> ,” Learning Trajectories for Robot Programing by Demonstration Using a Coordinated Mixture of Factor Analyzers”	27
3	SYSTEM ANALYSIS	29
	3.1 EXISTING SYSTEM	29
	3.2 DISADVANTAGE OF EXISTING SYSTEM	29
	3.3 PROPOSED SYSTEM	30
	3.4 ADVANTAGE OF PROPOSED SYSTEM	30
	3.5 APPLICATIONS	31
4	SYSTEM DESIGN	32

	4.1 SYSTEM ARCHITECTURE	32
	4.2 WORKING	32
	4.3 MODULE DESCRIPTION	33
	4.3.1 Power supply module	34
	4.3.2 Processing module	36
	4.3.3 Camera module	38
	4.3.4 Microcontroller	40
	4.3.5 USB to TTL converter	
	4.3.6 Motor driver	
	4.3.7 DC motor	
5	SOFTWARE DESCRIPTION	43
	5.1. Visual studio	43
	5.1.2 Python	44
6	RESULTS AND DISCUSSION	49
	6.1 Final results obtained	49
7	CONCLUSION AND FUTURE WORK	60
	7.1 CONCLUSION	60
	7.2 FUTURE WORK	60
	REFERENCES	61
	APPENDIX	64

LIST OF FIGURES

FIGURE NO	NAME OF THE FIGURE	PAGE NO.
1.1	OpenCV software architecture	4
1.2	Structure of embedded systems	6
1.3	MQTT architecture diagram	
4.1	Hardware architecture	13
4.2	Robotic architecture	14
4.3	Power supply	
4.4	Raspberry pi	
4.5	USB camera	
4.6	Atmega-328 architecture	
4.7	USB-TO-TTL converter	
4.8	Motor driver	
4.9	DC motor	
6.1	Complete hardware setup	32
6.2	Top view of the robotic setup	38
6.3	Login page	38
6.4	Home page	39
6.5	Patrol robot initiation	41
6.6	Abnormal condition page of the mobile application.	

LIST OF SYMBOLS AND ABBREVIATIONS

CNN	Convolutional Neural Network
RNNS	Recurrent Neural Network
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
ReLU	Rectified Linear Unit
SARSA	State-Action-Reward-State-Action
DDPG	Deep Deterministic Policy Gradient

CHAPTER -1

INTRODUCTION

1.1 GENERAL

While state police forces are responsible for maintaining law and order and investigating crimes, central forces assist them with intelligence and internal security challenges (e.g., insurgencies). Expenditure on police accounts for about 3% of the central and state government budgets.

State police forces had 24% vacancies (about 5.5 lakh vacancies) in January 2016. Hence, while the sanctioned police strength was 181 police per lakh persons in 2016, the actual strength was 137 police. Note that the United Nations recommended standard is 222 police per lakh persons.

- 86% of the state police comprises of constabulary. Constables are typically promoted once during their service, and normally retire as head constables. This could weaken their incentive to perform well.
- Crime per lakh population has increased by 28% over the last decade (2005-2015). However, convictions have been low. In 2015, convictions were secured in 47% of the cases registered under the Indian Penal Code, 1860. The Law Commission has observed that one of the reasons behind this is the poor quality of investigations.
- CAG audits have found shortages in weaponry with state police forces. For example, Rajasthan and West Bengal had shortages of 75% and 71% respectively in required weaponry with the state police.
- The Bureau of Police Research and Development has also noted a 30.5% deficiency in stock of required vehicles (2,35,339 vehicles) with the state forces.

- However, funds dedicated for modernisation of infrastructure are typically not utilised fully. For example, in 2015-16, only 14% of such funds were used by the states.

1.2 TECHNOLOGY USED

1.2.1 Robotics

Robotics is an interdisciplinary research area at the interface of computer science and engineering. Robotics involves design, construction, operation, and use of robots. The goal of robotics is to design intelligent machines that can help and assist humans in their day-to-day lives and keep everyone safe. Robotics draws on the achievement of information engineering, computer engineering, mechanical engineering, electronic engineering and others.

Robotics develops machines that can substitute for humans and replicate human actions. Robots can be used in many situations and for many purposes, but today many are used in dangerous environments (including inspection of radioactive materials, bomb detection and deactivation), manufacturing processes, or where humans cannot survive (e.g. in space, underwater, in high heat, and clean up and containment of hazardous materials and radiation). Robots can take on any form but some are made to resemble humans in appearance. This is said to help in the acceptance of a robot in certain replicative behaviors usually performed by people. Such robots attempt to replicate walking, lifting, speech, cognition, or any other human activity. Many of today's robots are inspired by nature, contributing to the field of bio-inspired robotics.

The concept of creating robots that can operate autonomously dates back to classical times, but research into the functionality and potential uses of robots did not grow substantially until the 20th century. Throughout history, it has been frequently

assumed by various scholars, inventors, engineers, and technicians that robots will one day be able to mimic human behavior and manage tasks in a human-like fashion. Today, robotics is a rapidly growing field, as technological advances continue; researching, designing, and building new robots serve various practical purposes, whether domestically, commercially, or militarily. Many robots are built to do jobs that are hazardous to people, such as defusing bombs, finding survivors in unstable ruins, and exploring mines and shipwrecks. Robotics is also used in STEM (science, technology, engineering, and mathematics) as a teaching aid.

1.2.2 OpenCV

OpenCV was started at Intel in 1999 by Gary Bradsky and the first release came out in 2000. Vadim Pisarevsky joined Gary Bradsky to manage Intel's Russian software OpenCV team. In 2005, OpenCV was used on Stanley, the vehicle who won 2005 DARPA Grand Challenge. Later its active development continued under the support of Willow Garage, with Gary Bradsky and Vadim Pisarevsky leading the project. Right now, OpenCV supports a lot of algorithms related to Computer Vision and Machine Learning and it is expanding day-by-day.

Currently OpenCV supports a wide variety of programming languages like C++, Python, Java etc and is available on different platforms including Windows, Linux, OS X, Android, iOS etc. Also, interfaces based on CUDA and OpenCL are also under active development for high-speed GPU operations.

OpenCV-Python is the Python API of OpenCV. It combines the best qualities of OpenCV C++ API and Python language.

OpenCV is the leading open source library for computer vision, image processing and machine learning, and now features GPU acceleration for real-time operation.

OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Adopted all around the world, OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 6 million. Usage ranges from interactive art, to mines inspection, stitching maps on the web or through advanced robotics.

OpenCV-Python

Python is a general purpose programming language started by Guido van Rossum, which became very popular in short time mainly because of its simplicity and code readability. It enables the programmer to express his ideas in fewer lines of code without reducing any readability.

Compared to other languages like C/C++, Python is slower. But another important feature of Python is that it can be easily extended with C/C++. This feature helps us to write computationally intensive codes in C/C++ and create a Python wrapper for it so that we can use these wrappers as Python modules. This gives us two advantages: first, our code is as fast as original C/C++ code (since it is the actual C++ code working in background) and second, it is very easy to code in Python. This is how OpenCV-Python works, it is a Python wrapper around original C++ implementation.

And the support of Numpy makes the task more easier. **Numpy** is a highly optimized library for numerical operations. It gives a MATLAB-style syntax. All the OpenCV array structures are converted to-and-from Numpy arrays. So whatever operations you can do in Numpy, you can combine it with OpenCV, which increases

number of weapons in your arsenal. Besides that, several other libraries like SciPy, Matplotlib which supports Numpy can be used with this.

So OpenCV-Python is an appropriate tool for fast prototyping of computer vision problems.

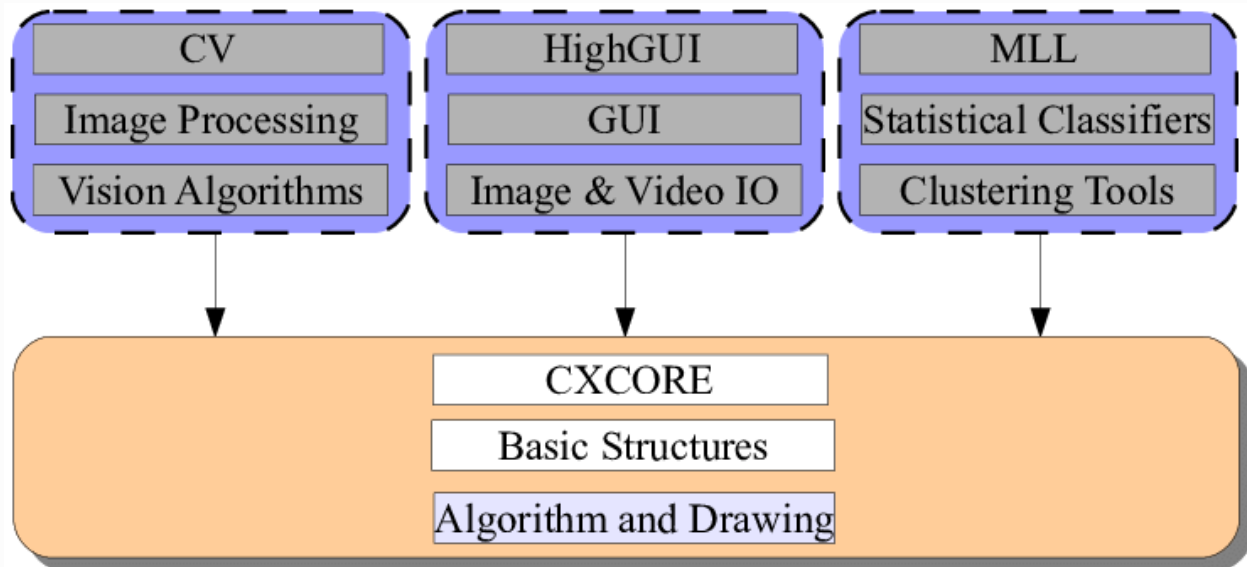


Figure 1.1 OpenCV software architecture

OpenCV Applications

OpenCV is being used for a very wide range of applications which include:

- Street view image stitching
- Automated inspection and surveillance
- Robot and driver-less car navigation and control
- Medical image analysis
- Video/image search and retrieval
- Movies - 3D structure from motion
- Interactive art installations

OpenCV Functionality

- Image/video I/O, processing, display (core, imgproc, highgui)
- Object/feature detection (objdetect, features2d, nonfree)
- Geometry-based monocular or stereo computer vision (calib3d, stitching, videostab)
- Computational photography (photo, video, superres)
- Machine learning & clustering (ml, flann)
- CUDA acceleration (*gpu*).

1.2.3 Embedded systems

An embedded system is a microprocessor-based computer hardware system with software that is designed to perform a dedicated function, either as an independent system or as a part of a large system. At the core is an integrated circuit designed to carry out computation for real-time operations.

Complexities range from a single microcontroller to a suite of processors with connected peripherals and networks; from no user interface to complex graphical user interfaces. The complexity of an embedded system varies significantly depending on the task for which it is designed.

Embedded system applications range from digital watches and microwaves to hybrid vehicles and avionics. As much as 98 percent of all microprocessors manufactured are used in embedded systems.

Embedded systems are managed by microcontrollers or digital signal processors (DSP), application-specific integrated circuits (ASIC), field-programmable gate arrays (FPGA), GPU technology, and gate arrays. These processing systems are integrated with components dedicated to handling electric and/or mechanical interfacing.

Embedded systems programming instructions, referred to as firmware, are stored in read-only memory or flash memory chips, running with limited computer hardware resources. Embedded systems connect with the outside world through peripherals, linking input and output devices.

1.2.3.1 Structure of embedded systems

The basic structure of an embedded system includes the following components:

Sensor: The sensor measures and converts the physical quantity to an electrical signal, which can then be read by an embedded systems engineer or any electronic instrument. A sensor stores the measured quantity to the memory.

A-D Converter: An analog-to-digital converter converts the analog signal sent by the sensor into a digital signal.

Processor & ASICs: Processors assess the data to measure the output and store it to the memory.

D-A Converter: A digital-to-analog converter changes the digital data fed by the processor to analog data

Actuator: An actuator compares the output given by the D-A Converter to the actual output stored and stores the approved output.

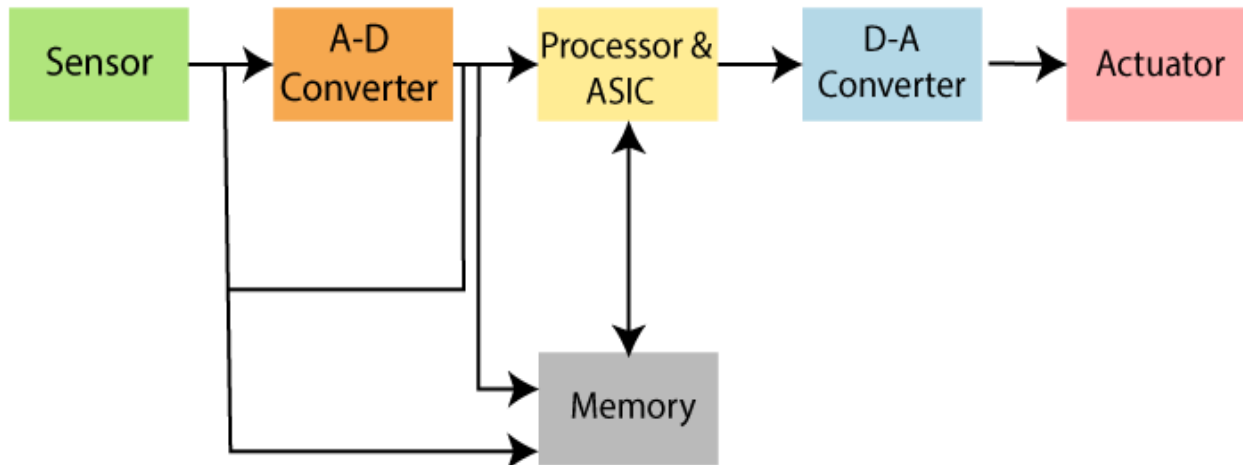


Figure 1.2: Structure of embedded systems

1.2.4 MQTT

MQTT(MQ Telemetry Transport or Message Queuing Telemetry Transport) is an open OASIS and ISO standard lightweight, publish-subscribe network protocol that transports messages between devices. The protocol usually runs over TCP/IP; however, any network protocol that provides ordered, lossless, bi-directional connections can support MQTT. It is designed for connections with remote locations where a "small code footprint" is required or the network bandwidth is limited.

The MQTT protocol defines two types of network entities: a message broker and a number of clients. An MQTT broker is a server that receives all messages from the clients and then routes the messages to the appropriate destination clients. An MQTT client is any device (from a micro controller up to a full-fledged server) that runs an MQTT library and connects to an MQTT broker over a network.

Information is organized in a hierarchy of topics. When a publisher has a new item of data to distribute, it sends a control message with the data to the connected broker. The broker then distributes the information to any clients that have subscribed to that topic. The publisher does not need to have any data on the number or locations

of subscribers, and subscribers, in turn, do not have to be configured with any data about the publishers.

If a broker receives a message on a topic for which there are no current subscribers, the broker discards the message unless the publisher of the message designated the message as a retained message. A retained message is a normal MQTT message with the retained flag set to true. The broker stores the last retained message and the corresponding QoS for the selected topic. Each client that subscribes to a topic pattern that matches the topic of the retained message receives the retained message immediately after they subscribe. The broker stores only one retained message per topic. This allows new subscribers to a topic to receive the most current value rather than waiting for the next update from a publisher.

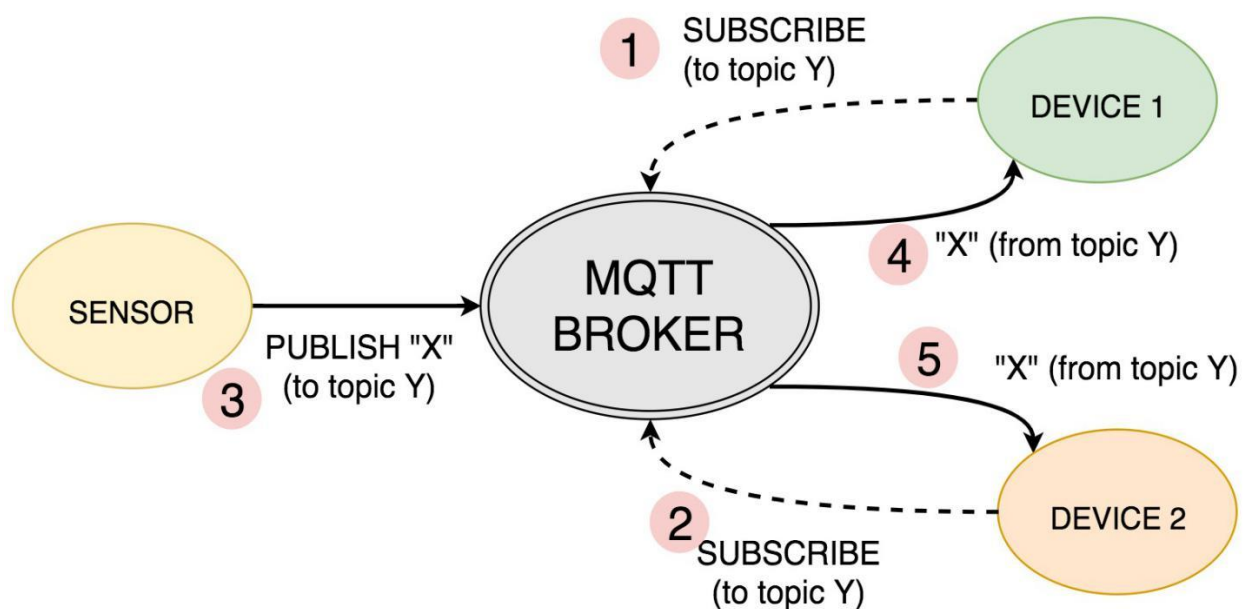


Figure:1.3 MQTT architecture diagram

When a publishing client first connects to the broker, it can set up a default message to be sent to subscribers if the broker detects that the publishing client has unexpectedly disconnected from the broker. Clients only interact with a broker, but a system may contain several broker servers that exchange data based on their current

subscribers' topics. A minimal MQTT control message can be as little as two bytes of data. A control message can carry nearly 256 megabytes of data if needed. There are fourteen defined message types used to connect and disconnect a client from a broker, to publish data, to acknowledge receipt of data, and to supervise the connection between client and server. MQTT relies on the TCP protocol for data transmission. A variant, MQTT-SN, is used over other transports such as UDP or Bluetooth. MQTT sends connection credentials in plain text format and does not include any measures for security or authentication. This can be provided by using TLS to encrypt and protect the transferred information against interception, modification or forgery. The default unencrypted MQTT port is 1883. The encrypted port is 8883.

1.2.5 React native app development:

Cross-platform development has become a great alternative to fully native mobile app development. Following the native mobile development approach, you create separate apps for Android and iOS. The cross-platform development allows you to cut expenses and save time by using the same code across both platforms. The React Native framework is a rising mobile solution and is considered the future of cross-platform mobile app development.

React Native is an open-source framework that allows to build a mobile app with only JavaScript. It was introduced by Jordan Walke, a Facebook software engineer, as a new technology for simpler development and a better user experience. The main distinctive of this framework is that React Native apps function just like native apps. They don't differ from apps built on Java, Objective-C or Swift and they use the same UI building blocks as native iOS or Android apps. But with React Native, building a mobile app is much faster and less expensive.

Reusable code & pre-built components

This is probably the most important advantage of React Native. With the convenience of this framework, developers don't need to build a separate mobile app for each platform, as almost 90 percent of React Native code can be reused between iOS and Android.

Simplified UI

React Native is all about the mobile UI. Comparing this framework to AngularJS or meteor JS, it looks more like a JavaScript library than a framework.

It's important to create a sequence of actions when building a mobile app, and React Native makes an implementing order just perfect. In addition, user interfaces designed in React Native are more responsive, decrease load time, and provide a smoother feel.

Third-party plugin support

The React Native framework is still advancing, so it might lack some components in the core framework. To fill this gap, React Native provides two types of third-party plugins: native modules and JavaScript modules.

1.3 OBJECTIVE:

- The objective of the project is to design a robot that can effectively patrol the streets.
- To develop a mobile application that can alert the nearby police personnel and provide live streaming.

CHAPTER -2

LITERATURE SURVEY

2.1 INTRODUCTION

The following shows survey on robots. The most popular of the existing techniques used for building of robots.

2.2 LITERATURE SURVEY

2.2.1 Haibing Wu , Bo Tao , Member, IEEE, Zeyu Gong, Member, IEEE, Zhouping Yin, Member, IEEE, and Han Ding, Senior Member, IEEE,” A Standalone RFID-Based Mobile Robot Navigation Method Using Single Passive Tag”

This article proposes a standalone radio frequency identification (RFID)-based mobile robot navigation method, in which a mobile robot equipped with reader antennas can be continuously guided to a static object marked with a single passive UHF RFID tag. An observation model based on the RFID phase difference is built and integrated into a particle filter, by which the instantaneous relative position between the mobile robot and the tagged object can be detected in real time. Based on the position information extracted from the RFID system, the mobile robot adjusts its pose to move toward the RFID-tagged object. Compared with the existing RFID-based mobile robot navigation methods, the proposed method requires no external sensors other than the RFID and requires only a single passive tag. Experiments using commercial off-the-shelf (COTS) RFID devices are performed, and the results indicate that the mobile robot can satisfactorily realize navigation task with a distance accuracy of 4.04 cm and a bearing accuracy of 2.23°. The proposed method is well applicable for the navigation scenes in which the absolute position of the tagged target object is not known beforehand.

Pros:

The method requires no external sensors other than the RFID and requires only a single passive tag.

Cons:

The system is not applicable for medical purposes, it cannot be used to treat patients.

2.2.2 Asif Khan, Bernhard Rinner, Senior Member, IEEE, and Andrea Cavallaro," Cooperative Robots to Observe Moving Targets: Review"

The deployment of multiple robots for achieving a common goal helps to improve the performance, efficiency, and/or robustness in a variety of tasks. In particular, the observation of moving targets is an important multirobot application that still exhibits numerous open challenges, including the effective coordination of the robots. This paper reviews control techniques for cooperative mobile robots monitoring multiple targets. The simultaneous movement of robots and targets makes this problem particularly interesting, and our review systematically addresses this cooperative multirobot problem for the first time. We classify and critically discuss the control techniques: cooperative multirobot observation of multiple moving targets, cooperative search, acquisition, and track, cooperative tracking, and multirobot pursuit evasion. We also identify the five major elements that characterize this problem, namely, the coordination method, the environment, the target, the robot and its sensor(s). These elements are used to systematically analyze the control techniques. The majority of the studied work is based on simulation and laboratory studies, which may not accurately reflect real-world operational conditions. Importantly, while our systematic analysis is focused on multitarget observation, our proposed classification is useful also for related multirobot applications.

Pros:

The system is useful also for related multi-robot applications.

Cons:

The system does not provide an option for remote controlled movement.

2.2.3 Shuai Li, Ruofan Kong, and Yi Guo, Senior Member, IEEE," Cooperative Distributed Source Seeking by Multiple Robots: Algorithms and Experiments"

We consider the problem of source seeking using a group of mobile robots equipped with sensors for source concentration measurement. In the formulation, the robot team cooperatively estimates the gradient of the source field, moves to the source by tracing the gradient-ascending direction, and keeps a predefined formation in movement. We present two control algorithms with all-to-all and limited communications, respectively. For the

case of all-to-all communication, rigorous analytic analysis proves that the formation center of the robots converges to the source in the presence of estimation errors with a bounded error, the upper bound of which is explicitly given. In the case of limited communication where centralized quantities are not available, distributed consensus filters are used to distributively estimate the centralized quantities, and then embedded in the distributed control laws. Numerical simulations are given to validate the effectiveness of the proposed approaches. Experimental results on the E-puck robot platform demonstrate satisfactory performances in a light source seeking application.

Pros:

Results on the E-puck robot platform demonstrate satisfactory performances in a light source seeking application.

Cons:

The system involves multiple robots, hence is not applicable.

2.2.4 Olzhas Adiyatov , Student Member, IEEE, Bexultan Rakhim, Altay Zhakatayev , Member, IEEE, and Huseyin Atakan Varol , Senior Member, IEEE,” Sensor Reduction of Variable Stiffness Actuated Robots Using Moving Horizon Estimation”

Variable stiffness actuated (VSA) robots are expected to play an important role in physical human–robot interaction, thanks to their inherent safety features. These systems can

control the position and stiffness concurrently by incorporating two or more actuators for each joint. Unfortunately, the need for extra sensors to measure the state of these actuators decreases the reliability of these systems. In this paper, we present a sensor reduction scheme for VSA robots. Specifically, we utilize moving horizon estimation (MHE) to estimate the unmeasured states of the system. Due to its ability to handle constraints, MHE is chosen as the estimation algorithm. The estimated states are then used by a nonlinear model predictive controller to implement a closed-loop control system. In order to show the efficacy of our framework, we conducted extensive simulation and realworld

experiments with a reaction wheel augmented VSA system. The objective of these experiments was to compare the control performance of the sensor reduced system (from four encoders to two encoders) with the system using the full set of states for control. The results of these experiments show the feasibility of the MHE-based sensor reduction. Sensor reduction might increase the reliability of VSA robots and might facilitate their

earlier introduction to the industrial environments.

Pros:

Produces very good accuracy compared to other state-of-the-art methods

Cons:

It inflexible for movement, hence is not feasible for large scale usage.

2.2.5 J. García, A. Bohórquez, and A. Valero," Suspension Effect in Tip-Over Stability and Steerability of Robots Moving on Sloping Terrains"

This article describes a model for a skid steer robot that quantifies the reaction forces between the wheels and the ground, which includes the effect of the passive suspension when the robot moves on flat and inclined terrains. In addition to the internal parameters of the suspension, this model included external parameters related to other components of robot and the robot terrain relationship. With this model, which was validated through

simulation, it was studied the effect that suspension produces in tip-over stability and steerability of robot when it moves on different terrain inclinations, specifically when the parameters of suspension system and wheel-ground contact are varied.

Pros:

The results are very satisfying, making it applicable for real time applications.

Cons:

The robot cannot be directed to move in the desired direction making it not feasible.

2.2.6 Milan Lukic, Ahmed Barnawi, Member, IEEE, and Ivan Stojmenovic, Fellow, IEEE,” Robot Coordination for Energy-Balanced Matching and Sequence Dispatch of Robots to Events”

Given a set of events and a set of robots, the dispatch problem is to allocate one robot for each event to visit it. In a single round, each robot may be allowed to visit only one event (matching dispatch), or several events in a sequence (sequence dispatch). In a distributed setting, each event is discovered by a sensor and reported to a robot. Here, we present novel algorithms aimed at overcoming the shortcomings of several existing solutions. We propose pairwise distance based matching algorithm (PDM) to eliminate long edges by pairwise exchanges between matching pairs. Our sequence dispatch algorithm (SQD) iteratively finds the closest event-robot pair, includes the event in dispatch schedule of the selected robot and updates its position accordingly. When event-robot distances are multiplied by robot resistance (inverse of the remaining energy), the corresponding energy-balanced variants are obtained. We also present generalizations which handle multiple visits and timing constraints. Our localized algorithm MAD is based on information mesh infrastructure and local auctions within the robot network for obtaining the optimal dispatch schedule for each robot. The simulations conducted confirm the advantages of our algorithms over other existing solutions in terms of average robot-event distance and lifetime.

Pros:

The simulations conducted confirm the advantages of our algorithms over other existing solutions in terms of average robot-event distance and lifetime.

Cons:

The system cannot be used for the betterment of general applications.

2.2.7 Jing Li; Qiang Huang; Zhangguo Yu; Xuechao Chen; Weimin Zhang; Huaxin Liu; Junyao Gao; Yingxian Duo," Integral Acceleration Generation for Slip Avoidance in a Planar Humanoid Robot"

Slip occurrence puts not only the humans but also the humanoid robots into dangerous situations. Although the detection and the estimation of slip have been well researched, studies on slip avoidance control in a humanoid robot are rare. Previous methods used to restrict the horizontal motion or regulate the vertical motion of the humanoid robot did not take the important rotational motion into account. This study proposes an integral acceleration generation method for slip avoidance that meets the real-time requirements, using the sagittal plane as an example. The proposed method makes full use of all the motions related to the slip in the sagittal plane to prevent the slip, including the translational and the rotational motions. The simulations and experiments on the physical humanoid robot demonstrate the effectiveness of the presented method.

Pros:

The proposed method makes full use of all the motions related to the slip in the sagittal plane to prevent the slip, including the translational and the rotational motions.

Cons:

Although the detection and the estimation of slip have been well researched, studies on slip avoidance control in a humanoid robot are rare.

2.2.8 Guillaume Doisy; Joachim Meyer; Yael Edan,” The Impact of Human–Robot Interface Design on the Use of a Learning Robot System ”

There is limited knowledge on how to design effective interfaces to enable nonexpert users to interact with robot learning algorithms. This paper focuses on an interface design challenge: How to provide the user with sufficient information about the learned behavior. A simulated robotic task where the robot has online learning capabilities was developed. This platform was used to study the impact of the variability of the environmental conditions, the information provided on the relation between the learned robot behavior and the conditions in the environment, and the presence of a preview of the learned robot behavior on the use of the system and on task performance. The results show significant effects of the type and the quantity of displayed information. Forty-two participants made the best use of brief and contextualized notifications about changes in the environment: their presence improved the overall performance and the usage of the automation and reduced the workload. In contrast, adding previews of the learned behavior surprisingly impaired performance and reduced the use of automation.

Pros:

In contrast, adding previews of the learned behavior surprisingly impaired performance and reduced the use of automation.

Cons:

The results are not accurate enough, making it not practical for real time applications.

2.2.9 Feng Gu; Yuqing He; Jianda Han,” Active Persistent Localization of a Three-Dimensional Moving Target Under Set-Membership Uncertainty Description Through Cooperation of Multiple Mobile Robots”

The persistent localization of moving targets is one of the most important applications of mobile robot systems. However, the sensors on each individual robot are often not sufficiently accurate for this task, presenting a severe limitation on the application of robots to many situations. To overcome this problem, systems of multiple robots have been established. These systems improve the localization accuracy of mobile targets by fusing together multiple sensing data. Numerous studies have demonstrated that the relative motion (or specifically, the relative pose) between the robot and the target, as well as that among different robots, strongly influences the final localization accuracy of the data fusion algorithm. In other words, the regulation of motion in a multiple-robot system will lead to better localization results. Thus, in this paper, a new active persistent localization (APL) scheme is proposed. The underlying concept of this scheme is based on set-membership descriptions of uncertainties. The basic problem is formulated in the framework of an enhanced set-membership filter, and a new data fusion algorithm with improved accuracy and convergence properties is designed. A motion-planning algorithm is then incorporated under the so-called optimal localization condition to complete the APL scheme. Simulations using multiple 3-D mobile robot systems and experiments on a multiple rotor-flying-robots test bed show that the proposed algorithm successfully enhances the accuracy of the persistent localization of the moving target.

Pros:

Experiments on a multiple rotor-flying-robots test bed show that the proposed algorithm successfully enhances the accuracy of the persistent localization of the moving target.

Cons:

The sensors on each individual robot are often not sufficiently accurate for this task.

2.2.10 C. Serkan Karagöz; H. Işıl Bozma; Daniel E. Koditschek,” Coordinated Navigation of Multiple Independent Disk-Shaped Robots”

This paper addresses the coordinated navigation of multiple independently actuated disk-shaped robots-all placed within the same disk-shaped workspace. Assuming perfect sensing, shared-centralized communications and computation, as well as perfect actuation, we encode complete information about the goal, obstacles, and workspace boundary using an artificial potential function over the configuration space of the robots' simultaneous nonoverlapping positions. The closed-loop dynamics governing the motion of each (velocity-controlled) robot take the form of the appropriate projection of the gradient of this function. We impose (conservative) restrictions on the allowable goal positions that yield sufficient conditions for convergence: We prove that this construction is an essential navigation function that guarantees collision-free motion of each robot to its destination from almost all initial free placements. The results of an extensive simulation study investigate practical issues such as average resulting trajectory length and robustness against simulated sensor noise.

Pros:

The results of an extensive simulation study investigate practical issues such as average resulting trajectory length and robustness against simulated sensor noise.

Cons:

The results are not up to realistic expectations.

2.2.11 Guanglong Du; Mingxuan Chen; Caibing Liu; Bo Zhang; Ping Zhang,” Online Robot Teaching With Natural Human–Robot Interaction”

With the development of Industry 4.0, robots tend to be intelligent and collaborative. For one, robots can interact naturally with humans. For another, robots can work collaboratively with humans in a common area. The traditional teaching method is no longer suitable for the production mode with human-robot collaboration. Since the traditional teaching processes are complicated, they need highly skilled staffs. This paper focuses on the natural way of online teaching, which can be applied to the tasks such as welding, painting, and stamping. This paper presents an online teaching method with the fusion of speech and gesture. A depth camera (Kinect) and an inertial measurement unit are used to capture the speech and gesture of the human. Interval Kalman filter and improved particle filter are employed to estimate the gesture. To integrate speech and gesture information more deeply, a novel method of audio-visual fusion based on text is proposed, which can extract the most useful information from the speech and gestures by transforming them into text. Finally, a maximum entropy algorithm is employed to deal with the fusion text into the corresponding robot instructions. The practicality and effectiveness of the proposed approach were validated by five subjects without robot teaching skills. The results indicate that the online robot teaching system can successfully teach robot manipulators.

Pros:

The results indicate that the online robot teaching system can successfully teach robot manipulators.

Cons:

The traditional teaching method is no longer suitable for the production mode with human-robot collaboration.

2.2.12 P.H.L. Silva; A.M.F. Burlamaqui,” System for Creation, Programming and Availability of Distributed Robot Teams using Collective Knowledge ”

This paper addresses the problem of programming, control and availability of distributed robots teams. Systems based on robots working together collaboratively has been the subject of much research in recent years. Because the rapid spread of technologies in mobile devices and Internet, as well as the benefits that robotics adds to the various everyday applications, the trend is that more and more it is present in various activities of human beings. For distributed robots systems there is a latent need for research to enable the use of such systems by end users with no or almost no experience with programming or robotics. Thus, this paper proposes a system architecture that allows the creation and programming of robots teams, allowing users to manage them as a single system and simple way using collective knowledge acquired both programmers as of the and users. The tests are performed by an application that was developed under the concepts of this architecture. It provides a set of special operations by which and users can connect, create and control robots teams at high level. The experiments use LEGO Mindstorms® robots as members of two robot teams. The process of creation, programming and control of the teams is presented in well defined steps. The results are satisfactory and show that the operating strategy of the proposed system meets fundamental aspects at the system level such as modularity, flexibility and ease of access for end users.

Pros:

The results are satisfactory and show that the operating strategy of the proposed system meets fundamental aspects at the system level such as modularity, flexibility and ease of access for end users

Cons:

The project is not feasible for automated robots.

**2.2.13 F.A.G. Becerra; A. B. Ortega; C. D. G. Beltrán; C. G. Valdivia; R.O.D. Arcega,”
Design and control of a new parallel robot for the rehabilitation of the hip-knee”**

The use of robots for passive medical rehabilitation has grown substantially in recent years due to the importance of achieving normality after having suffered an injury in any joint of the human body. In the present work we propose a new robot of rehabilitation for hip flexion movement that consists of two linear guide systems in the Z and Y axes. The kinetic analysis of the robot is made, proposing a new kinematic model of five Links. The geometric relationship between the linear guide and the flexion angle is used in such a way that it would be only necessary to measure the position of the linear guide without the measurement of the flexion angle of the hip. The mathematical model of the linear guides includes the dynamics of the CD motors, and a law of control GPI robust is proposed to control the position of the guides.

Pros:

The mathematical model of the linear guides includes the dynamics of the CD motors, and a law of control GPI robust is proposed to control the position of the guides.

Cons:

The robot positioning is not properly identified leaving more scope for improvement.

2.2.14 Takayuki Kanda, Member, IEEE, Dylan F. Glas, Masahiro Shiomi, and Norihiro Hagita, Senior Member, IEEE,” Abstracting People’s Trajectories for Social Robots to Proactively Approach Customers”

For a robot providing services to people in a public space such as a shopping mall, it is important to distinguish potential customers, such as window shoppers, from other people, such as busy commuters. In this paper, we present a series of abstraction techniques for people's trajectories and a service framework for using these techniques in a social robot, which enables a designer to make the robot proactively approach customers by only providing information about target local behavior. We placed a ubiquitous sensor network consisting of six laser range finders in a shopping arcade. The system tracks people's positions as well as their local behaviors, such as fast walking, idle walking, wandering, or stopping. We accumulated people's trajectories for a week, applying a clustering technique to the accumulated trajectories to extract information about the use of space and people's typical global behaviors. This information enables the robot to target its services to people who are walking idly or stopping. The robot anticipates both the areas in which people are likely to perform these behaviors as well as the probable local behaviors of individuals a few seconds in the future. In a field experiment, we demonstrate that this service framework enables the robot to serve people efficiently.

Pros:

The system provides a framework to assess engineering risk and the wider financial implications of the new technologies as they are introduced.

Cons:

The system is not recommended, the vehicles are not accurately recognized using this feature.

2.2.15 Matthew Field; David Stirling; Zengxi Pan; Fazel Naghdy," Learning Trajectories for Robot Programing by Demonstration Using a Coordinated Mixture of Factor Analyzers"

This paper presents an approach for learning robust models of humanoid robot trajectories from demonstration. In this formulation, a model of the joint space trajectory is represented as a sequence of motion primitives where a nonlinear dynamical system is learned by constructing a hidden Markov model (HMM) predicting the probability of residing in each motion primitive. With a coordinated mixture of factor analyzers as the emission probability density of the HMM, we are able to synthesize motion from a dynamic system acting along a manifold shared by both demonstrator and robot. This provides significant advantages in model complexity for kinematically redundant robots and can reduce the number of corresponding observations required for further learning. A stability analysis shows that the system is robust to deviations from the expected trajectory as well as transitional motion between manifolds. This approach is demonstrated experimentally by recording human motion with inertial sensors, learning a motion primitive model and correspondence map between the human and robot, and synthesizing motion from the manifold to control a 19 degree-of-freedom humanoid robot.

Pros:

This approach is demonstrated experimentally by recording human motion with inertial sensors, learning a motion primitive model and correspondence map between the human and robot, and synthesizing motion from the manifold to control a 19 degree-of-freedom humanoid robot.

Cons:

Accuracy is not up to the mark, leaving larger scope for improvement.

2.3 DISADVANTAGES IN EXISTING SYSTEM

- The system is RFID based making it not flexible for movement.

- The system reader antennas making it not feasible for extensive real time applications.
- The system is not applicable for medical purposes, it cannot be used to treat patients.
- The movement of the robot cannot be lively monitored or controlled.

CHAPTER - 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM:

This article proposes a standalone radio frequency identification (RFID)-based mobile robot navigation method, in which a mobile robot equipped with reader antennas can be continuously guided to a static object marked with a single passive UHF RFID tag. An observation model based on the RFID phase difference is built and integrated into a particle filter, by which the instantaneous relative position between the mobile robot and the tagged object can be detected in real time. Based on the position information extracted from the RFID system, the mobile robot adjusts its pose to move toward the RFID-tagged object. Compared with the existing RFID-based mobile robot navigation methods, the proposed method requires no external sensors other than the RFID and requires only a single passive tag. Experiments using commercial off-the-shelf (COTS) RFID devices are performed, and the results indicate that the mobile robot can satisfactorily realize navigation task with a distance accuracy of 4.04 cm and a bearing accuracy of 2.23°. The proposed method is well applicable for the navigation scenes in which the absolute position of the tagged target object is not known beforehand.

3.2 DISADVANTAGES OF EXISTING SYSTEM:

- The system is RFID based making it inflexible for movement.
- The system reader antennas making it not feasible for extensive real time applications.
- The system is not applicable for medical purposes, it cannot be used to treat patients.
- The movement of the robot cannot be lively monitored or controlled.

3.3 PROPOSED SYSTEM:

Robberies, burglaries and thefts continue to plague the length and breadth of this country. As per the National Crime Records Bureau (NCRB), 2,44,119 cases of robbery, theft, burglary, dacoity, among others, took place in residential premises in 2017. This was a jump of over 10% from 2016 when the number of such cases stood at 2,20,854. The financial loss due to these thefts and burglaries are staggering. In 2017, value of property stolen from residential premises was in excess of Rs. 2065 crores, a 40% jump from Rs. 1,475 crores stolen the previous year. Indeed, home safety is a serious concern for the residents as well as the police and the authorities. To overcome this problem we have developed a project that gives a technological solution using a robot. In this project we develop a robot with a hardware module connected with a camera and sound sensors embedded to it. A mobile application is developed using react native. The robot is given a specified distance to patrol around repeatedly, in case an abnormal noise is detected by the sound sensor the robot immediately alerts the nearby police personnel with an alert and also provides a live streaming through the mobile application.

3.4 ADVANTAGES:

- Reliable and effective solution for patrolling during the night.
- Reduces the manpower required for night patrolling.

- Effectively reduces crime
- A mobile application is developed to alert and provide the live streaming to the police personnel.

3.5 APPLICATIONS:

- Applicable for night patrolling
- Applicable in the police department
- Applicable for security purposes.

CHAPTER 4

SYSTEM DESIGN

DETAILED DESIGN OF THE PROJECT:

This chapter describes the overall and the detailed architectural design. It also describes each module that is to be implemented along with Data Flow diagram.

4.1 ARCHITECTURE DIAGRAM:

The architectural diagram is shown in Figure 4.1

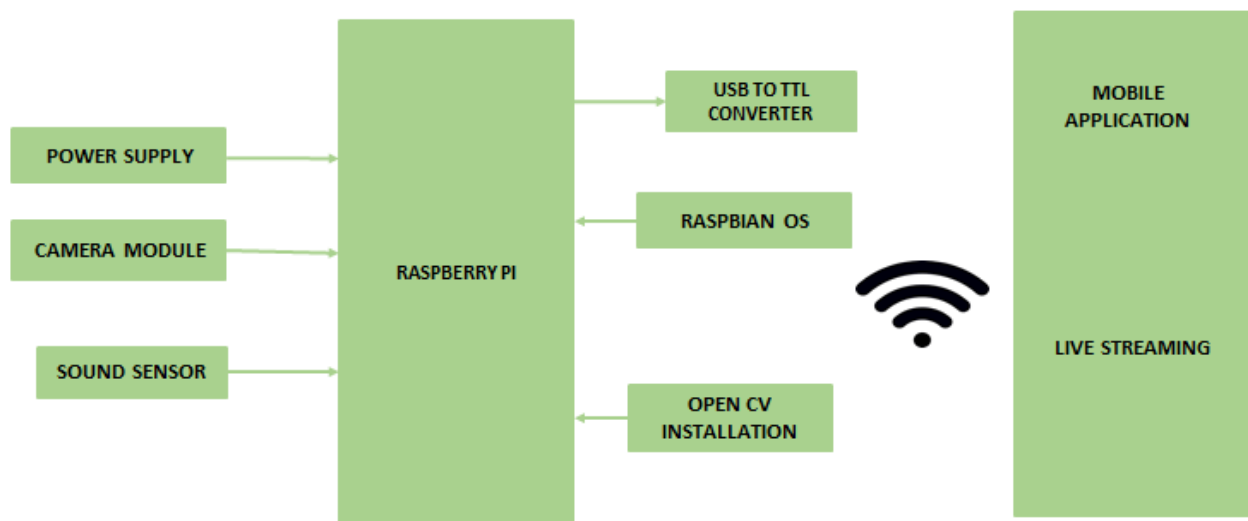


Figure 4.1 Hardware architecture

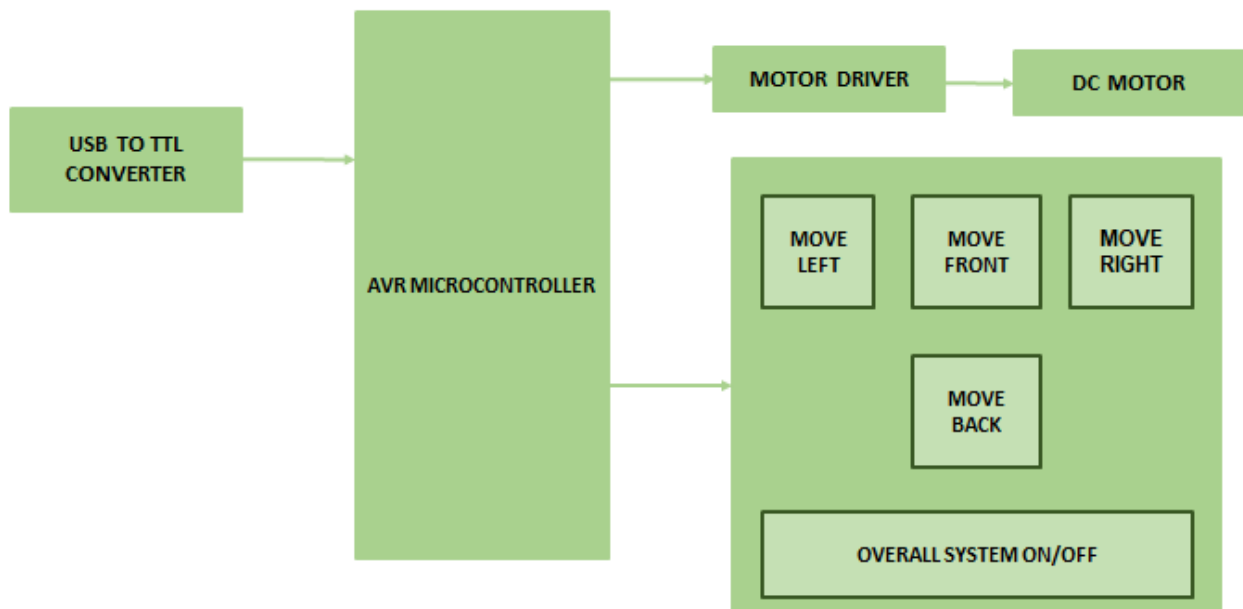


Figure 4.2 Robotic architecture

4.2 WORKING:

To overcome this problem we have developed a project that gives a technological solution using a robot. In this project we develop a robot with a hardware module connected with a camera and sound sensors embedded to it. A mobile application is developed using react native. The robot is given a specified distance to patrol around repeatedly, in case an abnormal noise is detected by the sound sensor the robot immediately alerts the nearby police personnel with an alert and also provides a live streaming through the mobile application.

4.3 MODULE DESCRIPTION:

- Power supply module

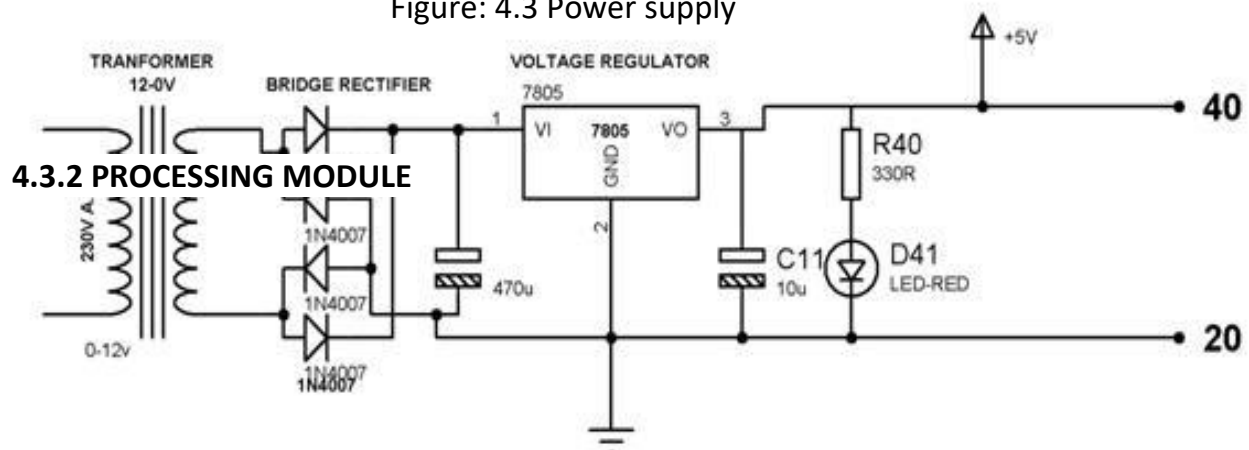
- Processing module
- Camera module
- Microcontroller
- USB to TTL converter
- Motor driver
- DC motor

4.3.1 POWER SUPPLY MODULE:

The power supply circuit consists of step-down transformer which is 230v step down to 12v. In this circuit 4 diodes are used to form bridge rectifier which delivers pulsating dc voltage & then fed to capacitor filter the output voltage from rectifier is fed to filter to eliminate any A.C. components present even after rectification.

The filtered DC voltage is given to regulator to produce 12v constant DC voltage. 230V AC power is converted into 12V AC (12V RMS value wherein the peak value is around 17V), but the required power is 5V DC; for this purpose, 17V AC power must be primarily converted into DC power then it can be stepped down to the 5V DC. AC power can be converted into DC using one of the power electronic converters called as Rectifier. There are different types of rectifiers, such as half-wave rectifier, full-wave rectifier and bridge rectifier. Due to the advantages of the bridge rectifier over the half and full wave rectifier, the bridge rectifier is frequently used for converting AC .

Figure: 4.3 Power supply



Raspberry Pi used for Maintenance and monitoring of attendance records and it is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation. The organization behind the Raspberry Pi consists of two arms. The first two models were developed by the Raspberry Pi Foundation. The Raspberry Pi hardware has evolved through several versions that feature variations in memory capacity and peripheral-device support.

The Raspberry Pi device looks like a motherboard, with the mounted chips and ports exposed (something you'd expect to see only if you opened up your computer and looked at its internal boards), but it has all the components you need to connect input, output, and storage devices and start computing.

Raspberry Pi is a low-cost, basic computer that was originally intended to help spur interest in computing among school-aged children. The Raspberry Pi is contained on a single circuit board and features ports for:

- HDMI
- USB 2.0
- Composite video
- Analog audio
- Power
- Internet
- SD Card

The computer runs entirely on open-source software and gives students the ability to mix and match software according to the work they wish to do.

The Raspberry Pi debuted in February 2012. The group behind the computer's development - the Raspberry Pi Foundation - started the project to make computing fun for students, while also creating interest in how computers work at a basic level. Unlike

using an encased computer from a manufacturer, the Raspberry Pi shows the essential guts behind the plastic. Even the software, by virtue of being open-source, offers an opportunity for students to explore the underlying code-if they wish.

The Raspberry Pi is believed to be an ideal learning tool, in that it is cheap to make, easy to replace and needs only a keyboard and a TV to run. These same strengths also make it an ideal product to jumpstart computing in the developing world. The quad-core Raspberry Pi 3 is both faster and more capable than its predecessor, the Raspberry Pi 2. For those interested in benchmarks, the Pi 3's CPU--the board's main processor--has roughly 50-60 percent better performance in 32-bit mode than that of the Pi 2, and is 10x faster than the

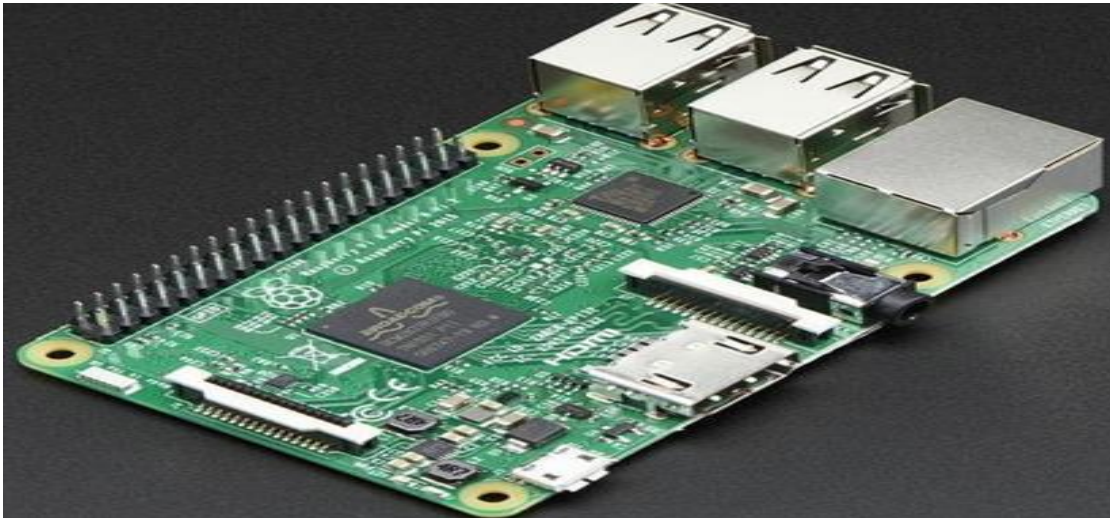


Figure: 4.4 Raspberry pi

Original single-core raspberry pi. Compared to the original Pi, real-world applications will see a performance increase of between 2.5x--for single-threaded applications--and more than 20x--when video playback is accelerated by the chip's NEON engine. Unlike its predecessor, the new board is capable of playing 1080p MP4 video at 60 frames per second (with a bitrate of about 5400Kbps), boosting the Pi's media center credentials. That's not to say, however, that all video will playback this

smoothly, with performance dependent on the source video, the player used and bitrate. The Pi 3 also supports wireless internet out of the box, with built-in Wi-Fi and Bluetooth. The latest board can also boot directly from a USB-attached hard drive or pen drive, as well as supporting booting from a network-attached file system, using PXE, which is useful for remotely updating a Pi and for sharing an operating system image between multiple machines.

4.3.3 CAMERA MODULE

An USB camera is a video camera that feeds or streams its image in real time to or through a computer to a computer network. When "captured" by the computer, the video stream may be saved, viewed or sent on to other networks travelling through systems such as the internet, and e-mailed as an attachment. When sent to a remote location, the video stream may be saved, viewed or on sent there. Unlike an IP camera (which connects using Ethernet or Wi-Fi), a webcam is generally connected by a USB cable, or similar cable, or built into computer hardware, such as laptops.

The term "USB" camera (a clipped compound) may also be used in its original sense of a video camera connected to the USB continuously for an indefinite time, rather than for a particular session, generally supplying a view for anyone who visits its web page over the Internet. Some of them, for example, those used as online traffic cameras, are expensive, rugged professional video cameras.



Figure: 4.5 USB camera

4.3.4 MICRO-CONTROLLER

ATmega-328 is basically an Advanced Virtual RISC (AVR) micro-controller. It supports the data up to eight (8) bits. ATmega-328 has 32KB internal built-in memory. This micro-controller has a lot of other characteristics. ATmega 328 has 1KB Electrically Erasable Programmable Read Only Memory (EEPROM). This property shows if the electric supply supplied to the micro-controller is removed, even then it can store the data and can provide results after providing it with the electric supply. Moreover, ATmega-328 has 2KB Static Random Access Memory (SRAM). ATmega 328 has several different features which make it the most popular device in today's market. These features consist of advanced RISC architecture, good performance, low power consumption, real timer counter having separate oscillator, 6 PWM pins, programmable Serial USART, programming lock for software security, throughput up to 20 MIPS etc. ATmega-328 is mostly used in Arduino.

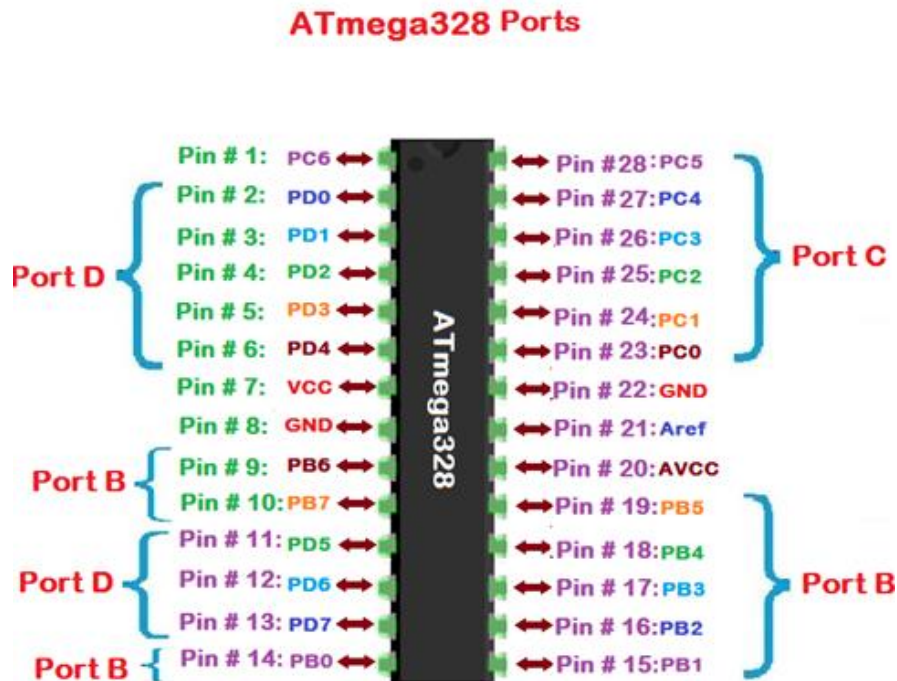


Figure: 4.6 Atmega-328 architecture

4.3.5 USB TO TTL CONVERTER

The USB TTL Serial cables are a range of USB to serial converter cables which provide connectivity between USB and serial UART interfaces. A range of cables are available offering connectivity at 5V, 3.3V or user specified signal levels with various connector interfaces. All cables feature an FTDI FT232R device integrated within the cable USB type 'A' connector, which provide access to UART Transmit (TX), Receive (Rx), RTS#, CTS#, VCC (5V) and GND connections. All cables are fully RoHS compliant and are FCC/CE approved. CP2102 chip from SiLabs is a single chip USB to UART bridge IC. It requires minimal external components. CP2102 can be used to migrate legacy serial port based devices to USB. Hobbyists can use it as a powerful tool to make all kinds of PC

interfaced projects. This module help all those who are comfortable with RS232/Serial Communication protocol, to build USB devices very easily.



Figure: 4.7 USB-TO-TTL converter

4.3.6 MOTOR DRIVER MODULE

L298N 2A Based Motor Driver is a high power motor driver perfect for driving DC Motors and Stepper Motors.

It uses the popular L298 motor driver IC and has an onboard 5V regulator which it can supply to an external circuit. It can control up to 4 DC motors, or 2 DC motors with directional and speed control.

This motor driver is perfect for robotics and mechatronics projects and perfect for controlling motors from microcontrollers, switches, relays, etc. Perfect for driving DC and Stepper motors for micro mouse, line following robots, robot arms, etc.

H-bridges are typically used in controlling motors speed and direction but can be used for other projects such as driving the brightness of certain lighting projects such as high powered LED arrays.

An H-Bridge is a circuit that can drive a current in either polarity and be controlled by *Pulse Width Modulation (PWM).

Pulse Width Modulation is a means in controlling the duration of an electronic pulse. In motors try to imagine the brush as a water wheel and electrons as the flowing droplets of water. The voltage would be the water flowing over the wheel at a constant rate, the more water flowing the higher the voltage. Motors are rated at certain voltages and can be damaged if the voltage is applied to heavily or if it is dropped quickly to slow the motor down. Thus PWM. Take the water wheel analogy and think of the water hitting it in pulses but at a constant flow. The longer the pulses the faster the wheel will turn, the shorter the pulses, the slower the water wheel will turn. Motors will last much longer and be more reliable if controlled through PWM.

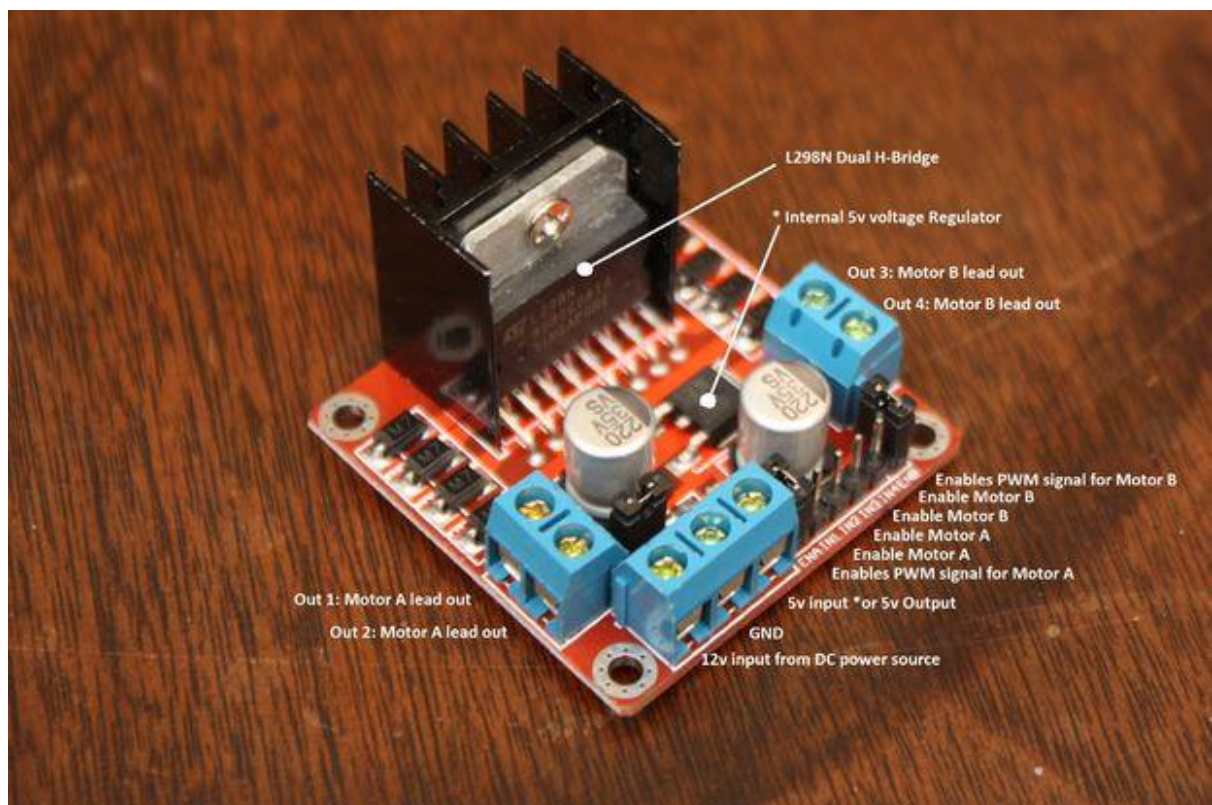


Figure: 4.8 Motor driver

4.3.7 DC MOTOR

An Electric DC motor is a machine which converts electric energy into mechanical energy. The working of DC motor is based on the principle that when a current-carrying conductor is placed in a magnetic field, it experiences a mechanical force. The direction of mechanical force is given by Fleming's Left-hand Rule and its magnitude is given by $F = BIL$ Newton.

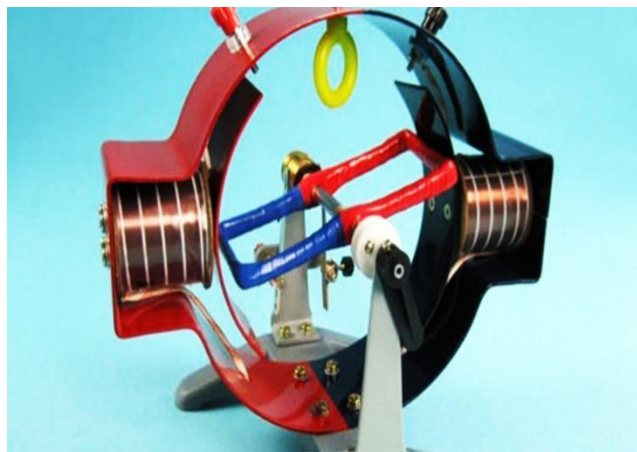


Figure: 4.9 DC motor

There is no basic difference in the construction of a DC generator and a DC motor. In fact, the same d.c. machine can be used interchangeably as a generator or as a motor. Like generators DC motors are also classified into shunt-wound, series-wound and compound-wound.

DC motors are seldom used in ordinary applications because all electric supply companies furnish alternating current.

However, for special applications such as in steel mills, mines and electric trains, it is

advantageous to convert alternating current into direct current in order to use dc motors. The reason is that speed/torque characteristics of d.c. motors are much more superior to that of a.c. motors.

CHAPTER 5

SOFTWARE DESCRIPTION

5.1 VISUAL STUDIO

In this project Microsoft visual studio is used as IDE.

Visual Studio Code combines the simplicity of a source code editor with powerful developer tooling, like IntelliSense code completion and debugging.

First and foremost, it is an editor that gets out of your way. The delightfully frictionless edit-build-debug cycle means less time fiddling with the environment, and more time executing on ideas.

Visual Studio Code supports macOS, Linux, and Windows - allowing the ground running, no matter the platform.

At its heart, Visual Studio Code features a lightning fast source code editor, perfect for day-to-day use. With support for hundreds of languages, VS Code helps in being instantly productive with syntax highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more. Intuitive keyboard shortcuts, easy customization and community-contributed keyboard shortcut mappings lets navigation of code with ease.

For serious coding, the benefit is from tools with more code understanding than just blocks of text. Visual Studio Code includes built-in support for IntelliSense code completion, rich semantic code understanding and navigation, and code refactoring.

And when the coding gets tough, the tough get debugging. Debugging is often the one feature that developers miss most in a leaner coding experience, so it is made to happen. Visual Studio Code includes an interactive debugger, so one can step through source code, inspect variables, view call stacks, and execute commands in the console.

VS Code also integrates with build and scripting tools to perform common tasks making everyday workflows faster. VS Code has support for Git so anyone can work with source control without leaving the editor including viewing pending changes diffs.

Allows customization of every feature to the liking of the user and install any number of third-party extensions. While most scenarios work "out of the box" with no configuration, VS Code also grows along with the user, and encourages user to optimize your experience to suit his/her unique needs.

VS Code includes enriched built-in support for Node.js development with JavaScript and TypeScript, powered by the same underlying technologies that drive Visual Studio. VS Code also includes great tooling for web technologies such as JSX/React, HTML, CSS, SCSS, Less, and JSON.

Architecturally, Visual Studio Code combines the best of web, native, and language-specific technologies. Using Electron, VS Code combines web technologies such as JavaScript and Node.js with the speed and flexibility of native apps. VS Code uses a newer, faster version of the same industrial-strength HTML-based editor that has powered the "Monaco" cloud editor, Internet Explorer's F12 Tools, and other projects. Additionally, VS Code uses a tools service architecture that enables it to integrate with many of the same technologies that power Visual Studio, including Roslyn for .NET, TypeScript, the Visual Studio debugging engine, and more.

Visual Studio Code includes a public extensibility model that lets developers build and use extensions, and richly customize their edit-build-debug experience.

5.2 ANDROID STUDIO:

In this project android studio is used as the IDE for mobile application development.

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains IntelliJ IDEA software and

designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development. To write some code for an Android application, some kind of environment or code editor is required that will make the task easy. Still code can be written in a normal text editor but that will increase the workload. So, Android Studio is an Integrated Development Environment (IDE) for Android app development. It is based on Intelligence IDEA. To enhance productivity while building Android application, Android Studio provides a number of features, They are

1. Instant run on a mobile device or any virtual device to check the working of the application
2. Extensive testing tools that can be used for better testing before launching an application on the Play Store
3. Flexible Gradle-based build system
4. Code autocomplete.

Database inspector: Enables a user to view the state of and make changes to the on-device database (using Jetpack Room or raw SQLite queries) in the IDE.

New project templates have been updated to use Material Design Components.

Integrated emulator window: The emulator can be run directly in the IDE instead of its own, separate window. Dagger/Hilt code navigation: Click on the new gutter actions to find out more about Dagger and Hilt types in your code.

ML Model Binding: Studio can generate code for an imported Tensor Flow Lite model which makes it easier to interact with that model from the app code.

5.3 EMBEDDED C

In this project the embedded C is used.

Embedded C is most popular programming language in software field for developing electronic gadgets. Each processor used in electronic system is associated with embedded software.

Embedded C programming plays a key role in performing specific function by the processor. In day-to-day life we used many electronic devices such as mobile phone, washing machine, digital camera, etc. These all device working is based on microcontroller that are programmed by embedded C.

Embedded C is perhaps the most popular languages among Embedded Programmers for programming Embedded Systems. There are many popular programming languages like Assembly, BASIC, C++ etc. that are often used for developing Embedded Systems but Embedded C remains popular due to its efficiency, less development time and portability.

Comments: Comments are readable text that are written to help us (the reader) understand the code easily. They are ignored by the compiler and do not take up any memory in the final code (after compilation).

There are two ways you can write comments: one is the single line comments denoted by `//` and the other is multiline comments denoted by `/*...*/`.

Preprocessor Directive: A Preprocessor Directive in Embedded C is an indication to the compiler that it must look in to this file for symbols that are not defined in the program.

In C Programming Language (also in Embedded C), Preprocessor Directives are usually represented using #include... or #define....

In Embedded C Programming, we usually use the preprocessor directive to indicate a header file specific to the microcontroller, which contains all the SFRs and the bits in those SFRs.

In case of 8051, Keil Compiler has the file “reg51.h”, which must be written at the beginning of every Embedded C Program.

Global Variables: Global Variables, as the name suggests, are Global to the program i.e. they can be accessed anywhere in the program.

Local Variables: Local Variables, in contrast to Global Variables, are confined to their respective function.

Main Function: Every C or Embedded C Program has one main function, from where the execution of the program begins.

CHAPTER-6

RESULTS AND DISCUSSIONS

6.1 FINAL OUTPUTS OBTAINED:

The below image shows the complete patrol robotic setup of the project:

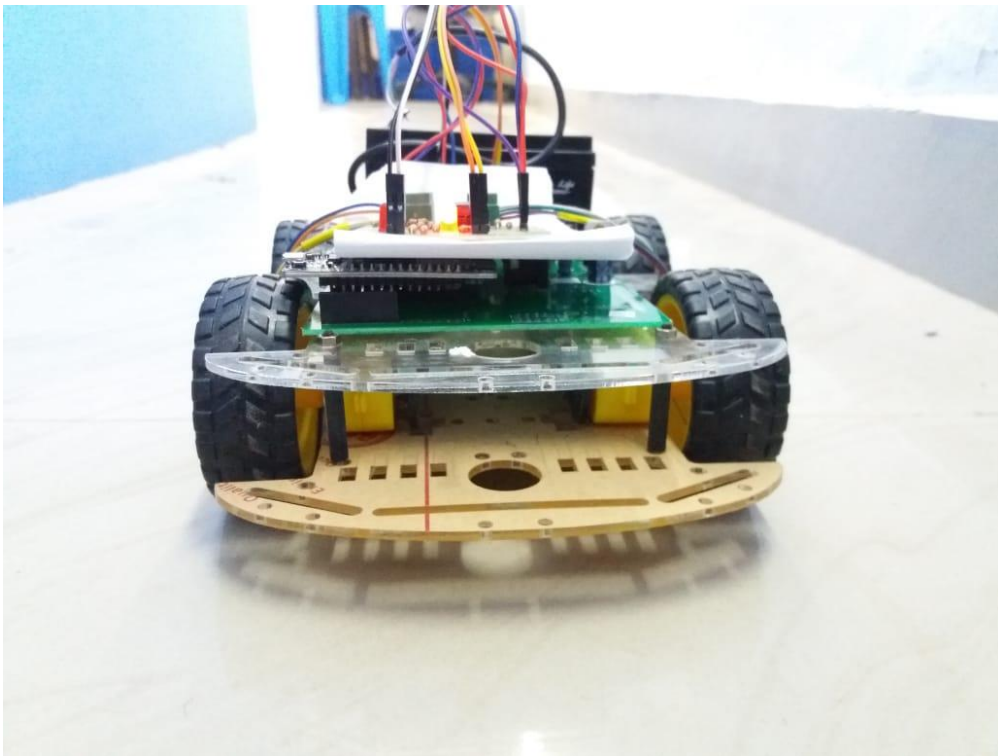


Figure 6.1: Complete hardware setup of the project

The below image shows the top view of the robotic setup:

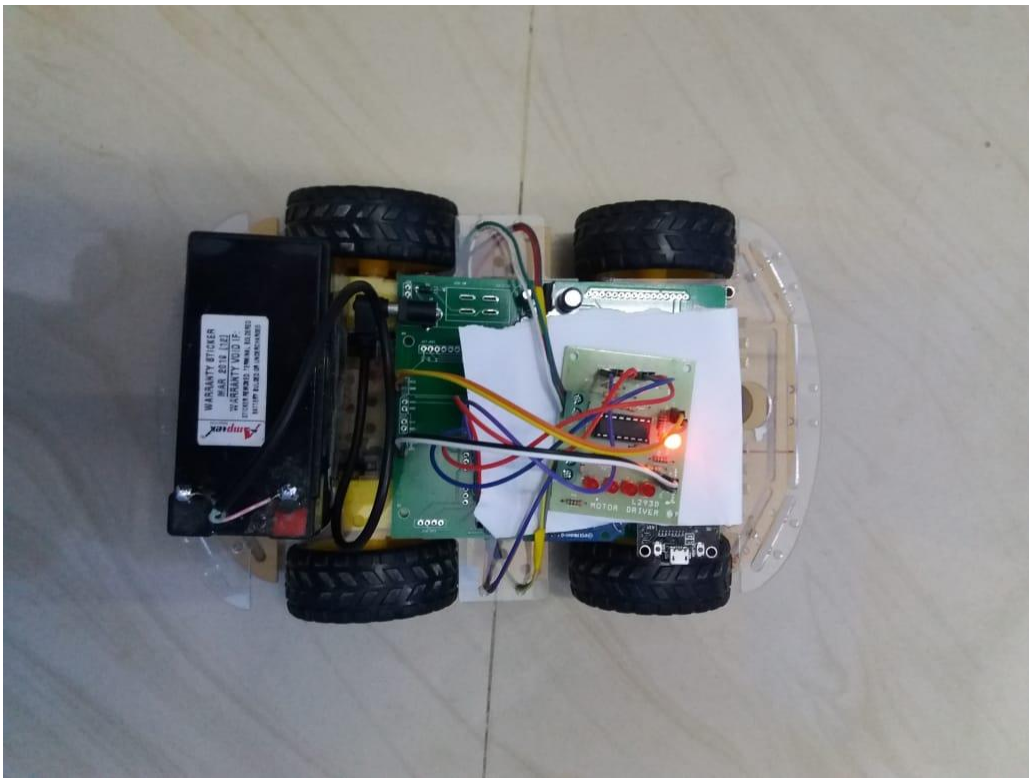


Figure 6.2: Top view of the robotic setup

A mobile application using a javascript framework react native is developed. The below image shows the login page of the mobile application:

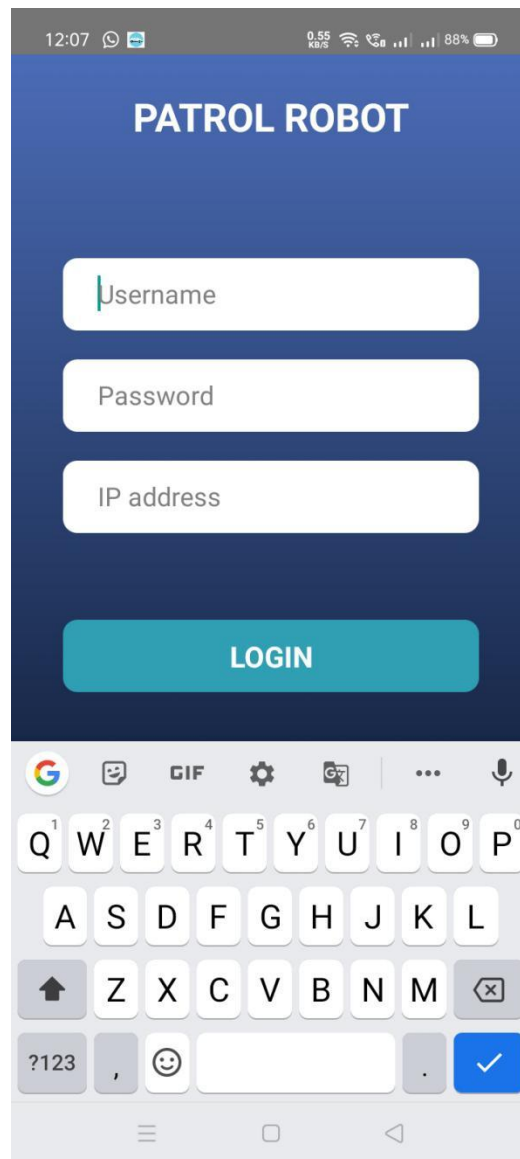


Figure 6.3: Login page

The below image shows the login page of the mobile application on successful login:



Figure 6.4:Home page

The below image shows the initiation of the robotic setup:

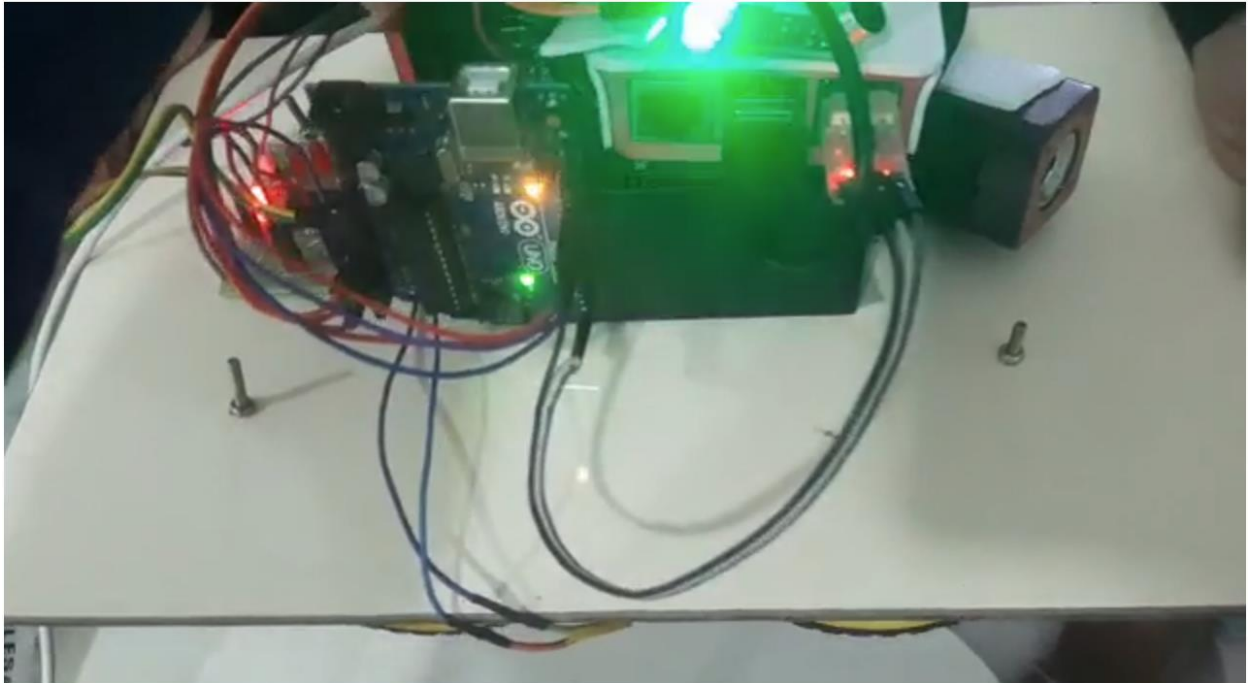


Figure 6.5: Patrol robot initiation

The below image shows the page of the mobile application on hearing an abnormal voice by the patrol robot with live streaming initiated and also a notification is sent through the mobile application:

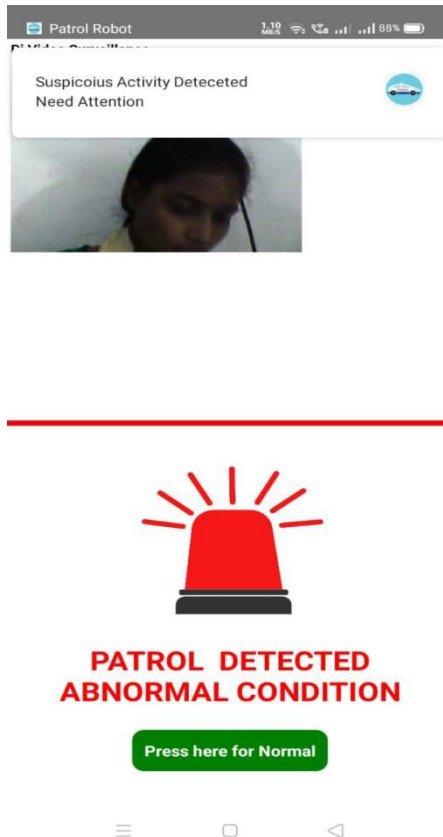


Figure 6.6: Abnormal condition page of the mobile application.

Thus, from the above results and discussion it is clear that the patrol robot is successfully developed and implemented with a mobile application developed using react native.

CHAPTER – 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSION

This project is used to effectively monitor and patrol the roads of highways with a robotic setup that has been effectively integrated to a mobile application. This also helps in providing efficient monitoring and patrol. If it is done manually which consumes more time and also involves human error rate. So, reduces the time required for manual patrolling and eliminates the human error rate by this project.

7.2 FUTURE WORK

In the coming future, we review the application of the patrol robot determine technology in the police department and it can promote for better patrolling of public spaces with more accuracy. In the police department there are more chance to develop or convert this project in many ways. Thus, this project has an efficient scope in coming future where manual patrolling of public spaces can be converted to computerized production in a cheap way.

.

REFERENCE:

- [1] Haibing Wu , Bo Tao , Member, IEEE, Zeyu Gong, Member, IEEE, Zhouping Yin, Member, IEEE, and Han Ding, Senior Member, IEEE,” A Standalone RFID-Based Mobile Robot Navigation Method Using Single Passive Tag”
- [2] Asif Khan, Bernhard Rinner, Senior Member, IEEE, and Andrea Cavallaro,” Cooperative Robots to Observe Moving Targets: Review”
- [3] Shuai Li, Ruofan Kong, and Yi Guo, Senior Member, IEEE,” Cooperative Distributed Source Seeking by Multiple Robots: Algorithms and Experiments”
- [4] Olzhas Adiyatov , Student Member, IEEE, Bexultan Rakhim, Altay Zhakatayev , Member, IEEE, and Huseyin Atakan Varol , Senior Member, IEEE,” Sensor Reduction of Variable Stiffness Actuated Robots Using Moving Horizon Estimation”
- [5] J. García, A. Bohórquez, and A. Valero,” Suspension Effect in Tip-Over Stability and Steerability of Robots Moving on Sloping Terrains”
- [6] Milan Lukic, Ahmed Barnawi, Member, IEEE, and Ivan Stojmenovic, Fellow, IEEE,” Robot Coordination for Energy-Balanced Matching and Sequence Dispatch of Robots to Events”
- [7] Jing Li; Qiang Huang; Zhangguo Yu; Xuechao Chen; Weimin Zhang; Huaxin Liu; Junyao Gao; Yingxian Duo,” Integral Acceleration Generation for Slip Avoidance in a Planar Humanoid Robot”
- [8] Guillaume Doisy; Joachim Meyer; Yael Edan,” The Impact of Human–Robot Interface Design on the Use of a Learning Robot System ”

- [9] **Feng Gu; Yuqing He; Jianda Han,**” Active Persistent Localization of a Three-Dimensional Moving Target Under Set-Membership Uncertainty Description Through Cooperation of Multiple Mobile Robots”
- [10] **C. Serkan Karagöz; H. Işıl Bozma; Daniel E. Koditschek,**” Coordinated Navigation of Multiple Independent Disk-Shaped Robots”
- [11] **Guanglong Du; Mingxuan Chen; Caibing Liu; Bo Zhang; Ping Zhang,**” Online Robot Teaching With Natural Human–Robot Interaction”
- [12] P.H.L. Silva; A.M.F. Burlamaqui,” System for Creation, Programming and Availability of Distributed Robot Teams using Collective Knowledge ”
- [13] F.A.G. Becerra; A. B. Ortega; C. D. G. Beltrán; C. G. Valdivia; R.O.D. Arcega,” Design and control of a new parallel robot for the rehabilitation of the hip-knee”
- [14] Takayuki Kanda, Member, IEEE, Dylan F. Glas, Masahiro Shiomi, and Norihiro Hagita, Senior Member, IEEE,” Abstracting People’s Trajectories for Social Robots to Proactively Approach Customers”
- [15] **Matthew Field; David Stirling; Zengxi Pan; Fazel Naghdy,**” Learning Trajectories for Robot Programing by Demonstration Using a Coordinated Mixture of Factor Analyzers”

APPENDIX:

SOUND CAPTURE:

```
import pyaudio

import wave

import soundfile as sf

import pyloudnorm as pyn

while True:

    CHUNK = 1024

    FORMAT = pyaudio.paInt16 #paInt8

    CHANNELS = 2

    RATE = 44100 #sample rate

    RECORD_SECONDS = 1

    WAVE_OUTPUT_FILENAME = "hum_ab40.wav"

    p = pyaudio.PyAudio()

    stream = p.open(format=FORMAT,

        channels=CHANNELS,

        rate=RATE,

        input=True,

        frames_per_buffer=CHUNK) #buffer
```

```

print("* recording")

frames = []

for i in range(0, int(RATE / CHUNK * RECORD_SECONDS)):

    data = stream.read(CHUNK)

    frames.append(data) # 2 bytes(16 bits) per channel

print("* done recording")

stream.stop_stream()

stream.close()

p.terminate()

# wf = wave.open(WAVE_OUTPUT_FILENAME, 'wb')

with wave.open(WAVE_OUTPUT_FILENAME, 'wb') as wf:

    wf.setnchannels(CHANNELS)

    wf.setsampwidth(p.get_sample_size(FORMAT))

    wf.setframerate(RATE)

    wf.writeframes(b''.join(frames))

    data, rate = sf.read("hum_ab40.wav")

    meter = pyln.Meter(rate) #

    loudness = meter.integrated_loudness(data)

    print(loudness)

wf.close()

```

ARDUINO_CODE:

```
char data;
```

```
int in1 = 4,in2 = 5,in3 = 12,in4 = 13;
```

```
void setup() {
```

```
    pinMode(in1,OUTPUT);
```

```
    pinMode(in2,OUTPUT);
```

```
    pinMode(in3,OUTPUT);
```

```
    pinMode(in4,OUTPUT);
```

```
    digitalWrite(in1,LOW);
```

```
    digitalWrite(in2,LOW);
```

```
    digitalWrite(in3,LOW);
```

```
    digitalWrite(in4,LOW);
```

```
    Serial.begin(9600);
```

```
    delay(1000);
```

```
    Serial.print("start");
```

```
}
```

```
void loop() { // READ DATA
```

```
if (Serial.available())  
  
  {  
  
    data = Serial.read();  
  
  }  
  
  if(data == '*')//forward  
  
  {  
  
    Serial.print("left");  
  
    digitalWrite(in1,LOW);  
  
    digitalWrite(in2,HIGH);  
  
    digitalWrite(in3,LOW);  
  
    digitalWrite(in4,LOW);  
  
  }  
  
  else if(data == '#')//backward  
  
  {  
  
    Serial.print("right");  
  
    digitalWrite(in1,LOW);  
  
    digitalWrite(in2,LOW);  
  
    digitalWrite(in3,HIGH);  
  
    digitalWrite(in4,LOW);  
  
  }  
  
  else if(data == '$')//right
```

```
{  
  
Serial.print("forward");  
  
digitalWrite(in1,LOW);  
  
digitalWrite(in2,HIGH);  
  
digitalWrite(in3,HIGH);  
  
digitalWrite(in4,LOW);  
  
}  
  
else if(data == '%')//left  
  
{  
  
Serial.print("backward");  
  
digitalWrite(in1,HIGH);  
  
digitalWrite(in2,LOW);  
  
digitalWrite(in3,LOW);  
  
digitalWrite(in4,HIGH);  
  
}  
  
  
else if(data == '~')//stop  
  
{  
  
  
  
Serial.print("stop");  
  
digitalWrite(in1,LOW);
```

```
digitalWrite(in2,LOW);  
digitalWrite(in3,LOW);  
digitalWrite(in4,LOW);  
}  
delay(100);  
}
```

ESP_CODE:

```
#include <ESP8266WiFi.h> // Enables the ESP8266 to connect to the local network (via WiFi)
```

```
#include <PubSubClient.h> // Allows us to connect to, and publish to the MQTT broker
```

```
const int ledPin = 2; // This code uses the built-in led for visual feedback that the button has been pressed
```

```
//const int buttonPin = 13; // Connect your button to pin #13
```

```
// WiFi
```

```
// Make sure to update this for your own WiFi network!
```

```
const char* ssid = "SIEORA";
```

```
const char* wifi_password = "9962037023";
```

```
// MQTT
```

```
// Make sure to update this for your own MQTT Broker!
```

```
const char* mqtt_server = "broker.hivemq.com";
```

```
const char* mqtt_topic = "robotic_car_project";
```

```
const char* mqtt_username = "robotic_car_project";
```

```
const char* mqtt_password = "robotic_car_project";
```

// The client id identifies the ESP8266 device. Think of it a bit like a hostname (Or just a name, like Greg).

```
const char* clientID = "robotic_car_project";
```

```
char motor;
```

```
// Initialise the Pushbutton Bouncer object
```

```
//Bounce bouncer = Bounce();
```

```
// Initialise the WiFi and MQTT Client objects
```

```
WiFiClient wifiClient;
```

```
PubSubClient client(mqtt_server, 1883, wifiClient); // 1883 is the listener port for the Broker
```

```
void ReceivedMessage(char* topic, byte* payload, unsigned int length) {
```

```
    // Output the first character of the message to serial (debug)
```

```
    Serial.println((char)payload[0]);
```

```
    //forward
```

```
    if ((char)payload[0] == '*' )
```

```
    {
```

```
digitalWrite(ledPin, LOW);

Serial.println("success");

Serial.write(payload[0]);

}

//backward

else if ((char)payload[0] == '#') {

    digitalWrite(ledPin, HIGH);

    Serial.write(payload[0]);

}

//right

else if ((char)payload[0] == '$') {

    digitalWrite(ledPin, LOW);

    Serial.write(payload[0]);

}

//left

else if ((char)payload[0] == '%') {

    digitalWrite(ledPin, HIGH);

    Serial.write(payload[0]);

}

//stop
```

```
    else if ((char)payload[0] == '~') {  
        digitalWrite(ledPin, HIGH);  
        Serial.write(payload[0]);  
    }  
}  
  
void setup() {  
    pinMode(ledPin, OUTPUT);  
    // pinMode(buttonPin, INPUT);  
    // Switch the on-board LED off to start with  
    digitalWrite(ledPin, HIGH);  
    delay(1000);  
    digitalWrite(ledPin, LOW);  
    delay(1000);  
    digitalWrite(ledPin, HIGH);  
  
    // Setup pushbutton Bouncer object  
    // bouncer.attach(buttonPin);  
    // bouncer.interval(5);  
  
    // Begin Serial on 115200
```

```
// Remember to choose the correct Baudrate on the Serial monitor!

// This is just for debugging purposes

Serial.begin(9600);

Serial.print("Connecting to ");

Serial.println(ssid);

// Connect to the WiFi

WiFi.begin(ssid, wifi_password);

// Wait until the connection has been confirmed before continuing


// Debugging - Output the IP Address of the ESP8266

Serial.println("WiFi connected");

Serial.print("IP address: ");

Serial.println(WiFi.localIP());

// Connect to MQTT Broker

// client.connect returns a boolean value to let us know if the connection was successful.

// If the connection is failing, make sure you are using the correct MQTT Username and
Password (Setup Earlier in the Instructable)

client.setCallback(ReceivedMessage);


if (Connect()) {

    Serial.println("Connected Successfully to MQTT Broker!");

}
```

```
else {  
    Serial.println("Connection Failed!");  
}  
}
```

```
void loop()  
{  
    client.loop();  
    delay(1000);  
}
```