DE-42 (MTS)

Aliza, Hassaan, Ibrahim, Komail

Design and Development of Commercial Surveillance Robot

COLLEGE OF
ELECTRICAL AND MECHANICAL ENGINEERING NATIONAL UNIVERSITY
OF SCIENCES AND TECHNOLOGY RAWALPINDI

2024

COLLEGE OF ELECTRICAL AND MECHANICAL ENGINEERING

DE-42 MTS
PROJECT REPORT

Design and Development of Commercial Surveillance Robot

Submitted to the Department of Mechatronics Engineering in

partial fulfillment of the requirements for the degree of

Bachelor of Engineering

in

Mechatronics

2024

Submitted By:

Aliza Naeem
Hassaan Ahmed Saifi
Ibrahim Nouman
Syed Komail Haider

Sponsoring DS:

Dr. Hassan Elahi (Supervisor)
Dr. Tahir Nawaz (Co-Supervisor)

CERTIFICATE OF APPROVAL

It is to certify that the project "Design and development of Commercial Surveillance Robot" was done by NS Aliza Naeem, NS Hassaan Ahmed Saifi, NS Ibrahim Noman and NS Syed Komail Haider under the supervision of Dr. Hassan Elahi and Dr. Tahir Habib Nawaz. This project is submitted to Department of Mechatronics Engineering. College of Electrical and Mechanical Engineering (Peshawar Road Rawalpindi), National University of Sciences and Technology, Pakistan in partial fulfillment of requirements for the degree of Bachelor of Engineering in Mechatronics Engineering.

Students:

1. **Aliza Naeem**

   NUST ID: 337922 Signature: ____

2. **Hassaan Ahmed Saifi**

   NUST ID: 359493 Signature: ____

3. **Ibrahim Nouman**

   NUST ID: 359584 Signature: ____

4. **Syed Komail Haider**

   NUST ID: 355429 Signature: ____

Approved By:

Project Supervisor: **Dr. Hassan Elahi**

Signature: ____

DECLARATION

We hereby declare that no portion of the work referred to in this Project Thesis has been submitted in support of an application for another degree or qualification of this of any other university or other institute of learning. If any act of plagiarism found, we are fully responsible for every disciplinary action taken against us depending upon the seriousness of the proven offence, even the cancellation of our degree.

1. **Aliza Naeem**

   NUST ID: 337922 — Signature: ________

2. **Hassaan Ahmed Saifi**

   NUST ID: 359493 — Signature: ________

3. **Ibrahim Noman**

   NUST ID: 359584 — Signature: ________

4. **Syed Komail Haider**

   NUST ID: 355429 — Signature: ________

COPYRIGHT STATEMENT

ACKNOWLEDGMENTS

We are grateful to the ALLAH Almighty who provides all the resources of every kind to us, sothat we make their proper use for the benefit of mankind. May Allah keep providing us with all the resources and the guidance to keep helping humanity.

We would like to start off by giving a heartfelt appreciation to our supervisors, Dr. Hassan Elahi, and Dr. Tahir Habib Nawaz. Their firm support, guidance and encouragement made us come a long way in this project.

Our journey together has been both intellectually enriching and inspiring, thanks to our teammates – Aliza Naeem, Hassan Ahmed Saifi, Ibrahim Noman and Syed Komail Haider –. Their diligence, cooperation, and unwavering support throughout this project have significantly contributed to our academic growth and created a truly memorable experience.

We would also like to express our gratitude to our families for their unwavering support and faith in our abilities. Their constant encouragement has been a driving force behind our efforts.

Finally, we are grateful to everyone who was involved in this project either directly or indirectly. Their contributions have been integral to the successful completion of our project.

Abstract

A robot is an autonomous electronic device that uses associated sensors to detect environmental changes. Robots are trained to respond when perceived values are less than or greater than a given value. The programmers begin by sketching a flowchart of their concept before creating the proper algorithms to lessen the complexity and number of iterations in their program. A robot's fundamental components include logic circuits, electronic elements, and programming. Robots have recently been designed to decrease the need for human resources and observation time in a burgeoning, quickly developing technology. Recently, many customers' interests in surveillance robots have grown for a variety of applications. In this project, a surveillance robot is created and constructed to move through a specific area, preferably a residential setting, record video and audio, and then transmit the data to the user over a network. In order to protect, act, and inform users, surveillance is the process of maintaining tight, systematic monitoring, observation, or supervision over a person or an area. Over a Wi-Fi network, users can use laptops or cellphones to communicate with the robot. Robotic surveillance systems can be used both indoors and outdoors by programming embedded devices. With the help of IoT technologies and the Raspberry Pi, the robot can be remotely controlled wirelessly. This sensor-based robot employs a Raspberry Pi for networking along with Ethernet, WiFi, and Bluetooth. Both autonomous and semiautonomous modes use Python as the programming language. Experimental tests on the built robot were conducted. Through a WiFi network, the end user successfully receives the recorded videos, photos, and audio. For particular purposes, the operator could steer and maneuver the robot in different directions. Users are drawn to the developed simple robot to record and transfer information for the safety.

SUSTAINABLE DEVELOPMENT GOALS

Considering all the possible sustainable development goals the following are directly/indirectly associated with our project:

SDG 08:

The functionality of the IoT-based surveillance robot project has a great contribution to SDG 8 aimed at the promotion of sustained, inclusive and sustainable economic growth, full and productive employment, and decent work for all. With its ability to provide effective and efficient monitoring, our robot makes workplaces more secure by preventing workers from theft and vandalism as well as other security breaches. This makes the foundation for productivity and economic activity to be safer. Further, the advancement and adoption of such cutting-edge technologies promote the growth of the tech sector and generation of new employment opportunities across the sphere, including robotics, software engineering, and cloud security. The use of surveillance software in the business reduces the time spent on operation and helps human resources to engage in high-value tasks. Our surveillance solution is also cost effective and efficient in supporting the small and medium enterprises as they watch over assets and businesses to ensure growth and sustainability especially in small businesses

SDG 11:

This project contributes to SDG 11 – it focuses on the safety and sustainable development of cities and human settlements. Using these robots to patrol cities will make cities safer and safer communities due to safer public areas and housing compounds which is beneficial to the lives of people in the city. The robot's connection to smart city initiatives aims to combine IoT-based technologies for effective city governance, as well as the efficiency and sustainability of urban services. Additionally, in disaster and emergency response settings, the robot can be used as source of monitoring and assessment on real time to support effective response and recovery for the benefit of urban resilience. Our robot can also sense the environment through various elements like air pollution or noise levels this robot is able to address environmental challenges associated with urban living.

TABLE OF CONTENTS

LIST OF FIGURES

STRUCTURE OF THESIS

This thesis is written in the following fashion:

Chapter 1: Provides the introduction about the project and what are the basic functionalities.

Chapter 2: Provides a Literature Review regarding the existing problems in the perception model and existing methodologies for developing an effective and robust fused model.

Chapter 3: Presents the details of methodology employed by our team in completing this project and explains the inner working of the project.

Chapter 4: Briefly discusses the summary of results and findings and comparison of accuracy with existing methodologies in terms of a more accurate depth perception model.

Chapter 5: Concludes the report and explores future possibilities and directions in which the project can be taken.

CHAPTER 1: INTRODUCTION

1.1 Overview:

The world has witnessed numerous inventions and this has been in the technological arena where the combination of Internet of things with robots can be seen as a modern-day approach to solving the world's challenges. Among all of these, the area of surveillance may be considered one of the most crucial fields in which innovation is not simply advisable, but mandatory. CCTV and other conventional security means are restrictive in terms of coverage, real-time analysis and timely delivery of information. But the emergence of 4G/5G-powered self-propelled surveillant robots that integrate modern functionalities such as face and object recognition, as well as obstacle detection and avoidance technology will change the security and monitoring game completely.

This thesis considers the idea generation, design, development, and deployment of a surveillance robot enabled with state of the art face detection and recognition, object detection and recognition, and obstacle avoidance services which are built using IoT technologies. This research aims at coming up with dynamic, efficient, and intelligent solution using the interconnectedness of IoT devices and mobility of robotic platforms Solution to Surveillance Issue Using Internet of Things and Robotic Platforms.

The combination of IoT technology with robotic technology paves the way for a number of opportunities. Transmitting its data to a human operator through networks, a surveillance robot becomes a sensing, central processing, and communication device. This helps not only in the situation update but also in the availability of information for subsequent decision-making and response. The feature of face detection and recognition makes it possible to define the persons under strict surveillance and recognize them, if they demonstrate aggressive gestures or words. Object detection and recognition enable the robot to recognize objects and classify them into different groups; this helps with the detection of anomalies and monitoring of the surrounding environment.

In addition, its incorporation of obstacle avoidance elements creates the required safety and effectiveness of the surveillance robot when in dynamic environments. With the help of sensors and a decision-making system, the robot would be able to analyze the situation and autonomously avoid obstacles to prevent possible collisions and therefore the delays in work.

This thesis is an attempt to understand the basic principles and technologies that make the IoT, robotics, and advanced surveillance systems mutually convenient. We consider the following design choices for developing an IoT surveillance robot, including the use of sensors and communications; navigation strategy; object and face detection; and integration platform. Moreover, we also explore the practical aspects and some of the ethical issues that would be connected to implementing such systems within real-life settings.

This research is not intended only to indicate a theoretical contribution to the realm of surveillance robotics, but also provide a workable solution to the field. Therefore, this thesis aims to contribute to filling the gap between theory and practice in the progress towards the full implementation and further development of next generation IoT powered surveillance robots with better detection and recognition abilities for creating safer and more secure and resilient communities.

1.2 Motivation:

The design, construction, and purpose behind this surveillance robot are prompted by an increased understanding of the growing requirements and concerns in the security and monitoring arena. The technology-driven surveillance approaches used to be prevalent in the past are insufficient to achieve the required level of success when dealing with current and newly emerging threats and risks. Traditional surveillance refers to plain cameras in motionless positions with human operational command centers that are impractical, impractical, and too slow to effectively address potential security hazards and other threats.

Against this backdrop, the motivation to create an IoT-enabled surveillance robot with advanced detection and recognition capabilities arises from a dual imperative: the urgency for increased security concerns & desire for the innovation-led products. The current forms of urbanization, globalization, and digitalization create never before experienced levels of security threats – from criminal activities to terrorist attacks to natural and medical emergencies. While dictating responses to these challenges, there is a critical need for surveillance systems that meet the described goals of being proactive, adaptive, intelligent, and efficient.

Further, there is the motivational aspect of the fact that IoT and robotics hold the potential to significantly impact the world of surveillance. The emergence of such technologies could present a revolution in how to address security issues. With the characteristics of powerful and highly integrated intelligence, through internal inter-connected sensors, big data collection and artificial intelligence, intelligent surveillance robots are capable of overcoming the deficiencies of traditional surveillance systems and developing the new generation of intelligent and conscious monitoring.

In addition, the main reason for the creation of this surveillance robot is based on the principle: there should be no evil technologies, the technologies must assist people and communities to live and prosper in safer and more secure environments. One of the specific improvements that can be achieved through the use of advanced features such as face and object detection & recognition and obstacle avoidance for the surveillance robot is the improvement on security, privacy, and ethical and inclusive institutional practices.

In sum, the desire to build this surveillance robot stems from the idea that innovation based on profound insight into the needs of society and technological opportunities can lead to the creation of such solutions that will significantly improve the current situation for all people. This project aims to offer a model of how a combination of IoT, robots, and surveillance can be used to address the threats and challenges to building a more resilient and connected world.

1.3 Problem Formulation:

The challenge is to create an IoT surveillance robot that includes the following features: autonomy; the ability to identify and track faces and objects; integration with existing IoT devices. This includes concerns about the reliability of the system as well as privacy and ethical issues that surround the work of surveillance teams. The goal of achieving these objectives is to develop a new service that will promote security measures, improve monitoring effectiveness and help to facilitate safer and more secure environments.

Previous Knowledge:

Robotics: Knowledge of the principles of robotics including hardware design,control systems and kinematics acquired during course work and practical.

Programming: Familiarity with programming languages like Python, C++, and Java to write the software components and algorithms for the surveillance robot.

IoT (Internet of Things): Knowledge of IoT concepts, protocols, and architectures and application of these concepts through coursework and practical projects to integrate the surveillance robot into IoT environments.

Computer Vision: Knowledge of computer vision such as image processing and object detection and recognition necessary for improving the identification of faces and objects for the surveillance robot.

Sensor Technology: The ability to evaluate systems that use sensor technologies for robots such as proximity sensors, cameras, and lidar to select appropriate sensors for obstacle avoidance and environmental monitoring.

Autonomous Systems: Ability in designing and programming navigation algorithms and decision making systems for the autonomous surveillance robot that operates in the dynamic environments.

Collaborative Projects: Working on cross-functional teams, developing skills of communication and collaboration, and project management skills, all of which are necessary for the development and implementation of the surveillance robot.

1.4 Contribution:

This project has great potential and it will benefit the society in a significant way in the following ways. The objective of the project is to develop a surveillance robot based on the Internet of things that focuses on detection and recognition and can be used in different surroundings to improve security features. Having this kind of advanced monitoring can effectively assist in preventing security breaches, which can include crimes as well as unsafe conditions or incidents. This is because the surveillance robot can supply actual time and drawn-in notifications to security experts and safety officers to assist in quicker decision-making intentions that may augment the overall safety and security of people and communities. Moreover, the project stresses the importance of privacy and responsible ethical matter in the process of conducting surveillance activities which would lead to the result of trust and respect within the population. In the end, use of such technologies not only improves security infrastructure but also ensures that more communities are made safer, resilient, and interconnected to ensure individuals can live and exist within their spaces in peace.

CHAPTER 2: LITERATURE REVIEW

A robot is an independent Mechanical machine that consists of its detectors to identify changes within the surrounding environment. Robots are trained specifically to reply in circumstances where perceived value is either less than or more than a certain amount. Initial, the conceptualization of the programmer involves drawing flowchart of their idea before developing the right algorithm to minimize the tediousness and iterations involved in a program. In focusing on the logic circuits of the robot, the electronic elements involved, and the programming aspect, one can describe with certainty the bare essentials of a robot. New robots have been made in recent times in order to reduce the extent of reliance on personnel and observation time which is essential in a growing, fast evolving sector. Currently, there is a giant shift in the customers' demands for surveillance robots because of the versatility it offers nowadays. In this project, a surveillance robot, of the writer's own design and construction, is to be built to physically navigate around a particular area, traditionally a home, record video and audio, and transmit the data back to the user through a network. Surveillance can be defined as the systematic, strict monitoring, supervision or observation of a person or an area through some actions, measures or ways in protection, action and information for the users. Where there is wireless connectivity, a user is able to converse with the robot through a laptop, cell phone, etc. Robotic surveillance systems can be fixed indoors and outdoors since the systems can be programmed using the devices that are incorporated into them. From this presentation, it also become evident that the robot can be remotely controlled wirelessly with the aid of IoT technologies and Raspberry Pi. This sensor-based robot carries out networking through Raspberry PI in addition to Ethernet, WiFi and Bluetooth. Every one of the both more independent as well as independant-driven ways of the requested programme employ Python as comprehending. Experimentation was also conducted on the robot which was constructed. Thus the end user gets the received videos, photos and audio captured into the gadget under a WiFi network. Other special intents could be assigned so that the operator could control and move the robot in one or another direction. The created simple robot attracts the users to record information, and transfer important information for security. [1]

2.1 Problem Statement:

To address the problems found in traditional surveillance systems and meet the ever-evolving demands of security and monitoring in various sectors, the development of a modern, innovative surveillance robot is vital for the complexity of weaved societies today. The main question that has to be solved is the development of a surveillance robot that could be equipped with a mechanism allowing it to move on various types of grounds, deliver real-time information and contribute to security enhancement, situation awareness, and management of incidents. Some of the main objectives important to be achieved within the framework are mobility and adaptability, sensory perception, autonomous and navigate, data processing and reporting, remote control, integration and connection, and to address ethical and safety considerations. Through designing and developing a new robot that provides the functionality that is missing between traditional approaches and modern demands, this work may revolutionize security and surveillance processes and guarantee protection for various spaces and groups.

2.2 HISTORICAL DEVELOPMENT

2.2.1) Early forms of surveillance robot:

Imagine a surveillance camera of the 80's in an ordinary convenient store. That enormous and expensive camera was connected to a VCR in a separate room by a wire that was inserted through the wall. In the proceeding decades more people use cameras because of these profound improvements of Cameras in terms of resolution, digital, storage and wireless transmission systems. However, having said this, it is evident that the social ramifications of being observed have not shifted; as a rule, when entering a business, we do not expect that the presence of video cameras will affect us. We expect people will be observing us closely and if we notice a camera, one can feel the moment uncomfortable and that is especially the case whenever we are doing something that we think will attract the attention of the observers. However, we usually come to the rational understanding that the videos, in which we participate, will be hardly studied or watched. Introduced as 'video analytics,' 'intelligent video analytics' or 'vCA,' new generation systems of capture-and-store video systems are slowly being incorporated into active monitoring technology. This way, computers will be able to understand the things and the actions that are on cameras top of just observing them.[2] This may be used when a person or an object is considered ‘suspicious,’ which might comprise of a call to the police. It can also be used to obtain confidential information relating to individuals or objects captured on the video for security or selling purposes. All the stupid video camera ‘eyes’ that follow around and watch every pour move we make will be accompanied by ‘brains’ that are smarter and smarter and recording us more and more. The video camera ‘eyes’ that we have seen in all these scenarios effectively monitor us, while ever smarter ‘brains’ monitor us discreetly. Want to get the sensation that technologists are training computers to do that monitoring in astounding methods through multiple podiums. The monitoring device turns active by beeping. Over the past couple of years, there has been an emergence of numerous cameras in our surrounding environment and the major part of recorded clips are never examined because the most part of the clips is rather uninteresting. Most cameras are also not monitored live; Bob Examples of this are words by security technologies for the Justice Department that have said, “the viewing of video screens is both monotone and riveting,” and “it is found that even after only twenty minutes of watch ". When it comes to overall monitor screens, it has been evident that length of focus has greatly reduced among people. The cameras we see today are still predominantly owned and operated by numerous small organizations, which means that it requires a great deal of one-on-one, repetitive work by individuals to decipher any video of ourselves (particularly videos of our mobility procured by multiple outlets). Although increasing numbers of our behaviors are being captured, they remain “practically invisible” in many cases Clearly, the cost of capturing and monitoring those behaviors is dropping, but tracking or analyzing them is still prohibitively costly.[3]

2.2.2) Evaluation from stationary cameras to mobile platforms:

2.2.2.1) Milestone in surveillance history:

It has been fifty years starting with the creation of the first surveillance robot. [4] Though, it would be unkind to limit a major event such as Shakey to the domain of security and surveillance However, it is illuminating to describe the background of the topic under discussion. Among the robots that were developed, Shakey a robot that was developed in 1960s was the first to have the ability to understand and make own sense of the surroundings it was placed in. It was invented at the Stanford Research Institute Int., by a team of engineers and headed by Dr.Charles Rosen – a legendary fellow hailed as one of the founders of artificial intelligence. With over 100,000 shares of stock and thousands of customers, the sales of Shakey marked the first time that robotics and artificial intelligence interests converged, although, formerly, they were only established individually. They first integrated these into a practical system in the first hardware called Shakey. The project was a formation of an enormous two-meter-tall device that was equipped with a TV camera, a telemetric camera to inspect the climate, and a radio antennae at its summit.

Figure 2.1 Shakey (1st mobile security robot)

It was founded by Defense Advanced Research Projects Agency known as DARPA that is responsible for the development of new technologies for the defense purposes. Randomly, I saw that Shakey steered itself and calculated how to get from point A to point B in the shortest time possible. For example, 'Search Algorithm A' was used, and it can be mentioned that it was utilized in the Curiosity rover that was sent to Mars in 2011; it could undoubtedly be considered one of the significant breakthroughs in robotics and artificial intelligence at the time. Shakey was the world's first deliberate, self-moving, perceiving, and decision-making robot. Even the first brick which should have been the most important, had been thrown down. [4]

Modern surveillance robots are fitted with high-end features like camera, motion sensor, temperature probe to not only detect threats but also the nature of the terrain they are navigating through plus the obstacles in their path.

2.2.2.2) The surveillance robot's early years:

Work done in 2016 aware S. A Joshi, A. Tondarkar et,al introduced a new surveillance system on borders/ remote place using multifunctional robots intending IoT current technology. These robots were used to sweep through the area in search of the enemy, record the event in a video camera and relay the signal directly to people who were allowed to watch the event live. Surveillance plays a rudimentary task when it is operating on the bordering area, thus, there exists a robot which may be utilized in the surveillance.

According to the article published by H. Kavalionak and C. Gennaro in 2017, systems of video surveillance as an imperatively required means of ensuring safety and regulating the space of private and public premises. They suggested the new way of a system of distribution face recognition protocol, using the computational resources of surveillance devices (cameras) to make the identification of people. Their works recorded that it can cut 50% load from the host server without negative ramifications on the quality of the surveillance service.

Z. Shaikh, et al. described another model that involves capturing pictures, and videos at real time times with a view of offering surveillances to specific districts or people. For the continuous processing and transmission of data, a Raspberry Pi processor was incorporated in the robotic system and was interfaced through ZigBee network system. These benefits become evident in experiments conducted in a military region or as a tool to conduct surveillance over a specific region or a person of interest during a fight or a hostage crisis scenario in a hostile environment.

Alvi M F, et al proposed a surveillance robot called Robot-cop where this center can be controlled by a webpage and can be remotely accessed. The Robot-cop is designed for surveillance and enforcement of space using video streaming option to avoid involvement of a police officer once deployed to a particular area. It needs to be able to cope with shocks and pressures at work or the ability to fall from a certain height and the ability to stabilize in the worst-case scenario, that is, to balance when landing in the wrong position.

In 2019, P. S Suryavamsi & A. A. Selvakumar proposed a light-weighted, user-controlled, smart surveillance robot applying to modern society to replace the limitations of the CCTV system. It simply moves around the house avoiding obstacles, and perhaps monitor the status of all cylinders including detecting gas leakage or the presence of burglars near the LPG cylinders and send all the monitored data to Thing-Speak IoT Cloud Server.

Ahamed Yasar and Dinesh Kumar R and others presented an idea of border monitoring system with the help of computer vision in the year 2020. One requires a camera, raspberry pie, arduino uno, and a buzzer, Open CV software detects positive and negative things and faces by using the gathered information/records as input on the raspberry pie. If the Pi module recognizes an object or a face that is not quite right or does not match the preset parameters, a signal is sent from the Raspberry Pi, which enables particular signals to be sent to a head or control room. [5]

2.2.2.3) Robotic surveillance is now a reality:

In the future, fifty years from now, we discover Yellow, a robot dog with artificial intelligence that interfaces with other high-tech security systems and has a variety of features and services.

Figure 2.2 Yellow, a robot dog with artificial intelligence

The Yellow robot, produced by the business Poseur Security and displayed at several events in 2022, has the following characteristics:The Yellow robot, produced by the business Poseur Security and displayed at several events in 2022, has the following characteristics:

1. We can greatly estimate the speed and minimum response time that 5G technology provides thanks to which the Internet of Things will get rid of existing models and the threat warning to the control center is minimized.
2. Prescriptive Analytics tools that can locate and flag materials that require attention based on applying intelligent algorithms. that help the robots to acquire information about the environment they are located in during their movement, for instance, temperature, or the presence of gases, and which can lead to fires or explosions. concerning security they are led to minimize dangers of harm causing.
3. It can move on any surface including on areas of rough surface and on steps.
4. The RB-WATCHER is another success story in the modern robotic security in contemporary robotics., The invention was developed by Robotnik , a Spanish based firm dealing in robotic goods and R &D projects aimed at developing products such as the RB-WATCHER. These are the descriptions of the RB-WATCHER: ‘‘An Autonomous mobile robot for security and surveillance in indoor and outdoor domain’.
5. Delightful from a robot-like RB-WATCHER, several key business benefits that a business can realise from these intelligent autonomous devices are:

1. **Always in action**: Prohibitionism of the fatigue is without an ounce of doubt an enormous boon of robot compared with the human work. Further, it is mentioned that the need for breaks is not necessary; therefore, observation may proceed without intermission. This ensures persistence of safety and minimization of precariously attached hazards.
2. **Recording and Evaluation of data**: In fact, actual-time surveillance robots are capable of recording and memorizing a lot of data. Afterward, all this data is scrutinized in order to identify some patterns and obtain optimization of security measures. That is, patterns may be built by accumulating other instances of anomaly and occurrence with a view to

identifying similar anomalies and perhaps preventing them from occurring at all. They have got a ‘crystal ball’ and can often ‘look ahead’.

3. **Hazard Avoidance:** The ability of surveillance and security robots to predict hazards in a way that is impossible for humans may seem obvious, but it never hurts to be reminded of it.
4. **Swiftness and Accuracy:** When predicting danger is not an option, only immediacy is left. Robotic surveillance systems because of its portability and ability to communicate with the control room.

2.2.3) Robotic security and surveillance: Ethical Issues

Any DL approach to scale information gathering is questionable for its ethical implications. This is evident in a very glaring instance which shows that SIF is not as efficient as it should be. We move to Meiyuan District in Beijing(Tokyo). A human-like security robot with a height of 5’ 7’’, standing with two legs and face recognition capability was developed by the Beijing Aerospace Automatic Control Research Institute (BAACI), and the robot has been constantly patrolling on streets round the clock. It seems to utilize a number of technologies such as environmental sensors, big data processing and biometric analytics. It will beep in case a person approaches it or touches it in a way that is considered problematic by the society. The services that the robot offer are not limited to allowing the guests to choose their meals and entertain them, but also offer them mundane things such as the weather forecast and music. [4]

Like every creature, all these varieties are not free of error. It is time to return to China to tell a story that caused quite a stir and scores citizens on the point system. Some things such as who is entitled to do particular things like flying are made possible by something called social credit. Facial recognition cameras can be installed in such places as airports where a set of intelligent cameras can track and recognize any individual. They are capable of denying them an entry into an airplane in case they do not have sufficient social credits.

<u>The Dong Mingzhu case</u>

Let me affirm, the facial recognition system how flexible the system is in identifying people and linking them to common events is demonstrated by an interesting fact. Currently, the spokeswoman of GREec Electric, a Chinese company that deals in household appliances, is the actress Dong Mingzhu. The image of Mingzhu was published in an advertizing manner on the side of a bus. The innocence of the actress was tainted when she was caught on ‘tape’ violating the traffic rule that prohibited passing through the red light. Of course, this must have been the invisible woman, the embodiment of a particular type or status rather than the individual as a living person. But, the camera failed to capture that moment. Once at the airport, the young lady was subjected to a momentary stop but it was all a misunderstanding and a fix was made immediately. However, there remained room for concern: And what if this person was not very popular and talented, but simply an ordinary man, who once made this mistake? Security bots’ utilization in purchasing personal data is not the only ethical problem at the heart of the matter. These gadgets are often purposely made outdated, and thus vulnerable to hacking by hackers and other malicious individuals. The particularly infamous robots are the ones that have not updated their security firmware thus making them very dangerous machines anyone with an intention of breaking into them can easily get an entry point. [4]

2.3) APPLICATIONS OF SURVILLANCE ROBOTS

2.3.1) SECURITY AND LAW ENFORCEMENT

1. Use of surveillance robot in broader security:

Surveillance is among the most important and sensitive responsibilities with a direct bearing on defense, which entails data capturing and processing in real-time in order to protect the citizens and act accordingly. Mainly, human monitoring is often carried out in the sensitive areas in respect with war situations, although there is always personnel loss. Due to technological developments, there is a possibility of having social importance areas that are monitored by robots and not human beings. These semi autonomous robots are capable of reading and observing small things as well as following instruction from the man. It can patrol borders and even help save lives and has the benefits of not eating, sleeping or feeling its emotions. [6] These adversary-controlled robots can also travel through rough and soggy surface and convey recordings of adversary undertakings. Self-directed robots shall in the future perform the roles of security guards and replace industrial workers and soldiers in some cases where access is restricted. These robots can as well offer security and perform the image and functions of security officers and can be a better tool in the protection of civilians during wars with less life dangers as compared to the traditional security officers.

Figure 2.3 Surveillance Robot Patrolling US boarder

2. **Urban policing and crowd management**:

The use of police robots and crowd control robots has several advantages for society, as the processes can become automated and robotized. Robots are not assured biological entities and hence they ought to be programmed to work under conditions with high risk of harm or death to man. Positive results can be obtained in cases when the situation is critical because of the usage of explosive materials, material objects, chemical aggrandizements, or industrial mishaps. In the pandemic period robots operating with the pandemics is in alignment with the humanitarian goals of using a robot; rather than, military and healthcare assisting robots or autonomous cars. The shortages and shortages of medical and law enforcement personnel can also be addressed by other kinds of problems for robotic AI systems. Detection systems based on AI can handle issues such as the predictability of crowd distribution and unpredictability of human movements. When the algorithmic and technical problems are fine-tuned, then robotic AI can certainly provide greater safety, openness, and efficiency as compared to the present scenario that often tends to incorporate force and violence. Robots on the traffic police: traffic police robots are the illustration of how robots can be successfully used in dangerous areas such as the traffic police stations where human traffic officers take a lot of risks and work in the environment which has much pollution.[7]

3. **Search and resource operations**:

Some recent disasters a few of these include the recent earth quakes in Nepal, typhoon Haiyan or the recent floods that occurred in Europe are and show how hard they are to manage by the local authorities and the emergency services. These occurrence affects the society in as much as it entailing high costs in terms of human lives and finances. Fire and rescue service are important in big disasters, and robot nodes can act as intelligent data collectors in big disasters. Such systems record accurate information that is time-stamped and geographically referenced, making investigations less complicated by assisting victims and insurance companies in providing proof and an easier process for authorities.[8]

2.3.2) INDUSTRIAL AND COMMERCIAL SECTOR

1. **Surveillance in manufacturing facilities**:

The application of robotics and automation in manufacturing is generally well-utilized by mobile robots in manufacturing plants for surveillance purposes. These robots can boost security, surveillance, improve operation since they can be equipped with different types of sensors, cameras. It is crucial to consider such aspects as the choice of a robot, the sensor suite, the camera systems, the identification and recognition of objects, navigation and mapping, remote monitoring, the detection of anomalies, integration with security systems, data storage/analysis, cooperation between human operators and robots, potential privacy issues, maintenance and technical support, considering the possibility of expansion, as well as developing the necessary skills and competencies in the operators. The robots employed in the manufacturing processes should be capable of functions such as familiarization with the manufacturing environment and its ability to maneuver as well as handle the terrain of the environment. Cameras are indispensable to the basic sensor suite, as is LDR, ultrasonic sensors, and infrared sensors. It is recommended to use high-performance video systems with motion-sensitive PTZ cameras in specific areas or at a scene.[9]

2. Inventory management and theft prevention:

Some of the specific tasks include: Security robots at the facilities are used to monitor the environment and the activities going on in the facilities through real time data collection. These findings can be used once again at a later time in an assessment of current and transforming trends that writers anticipate in the area of security policies and measures. Adopting informed decisions about the security of the company can be helped by information collected by robots. This article mentions that one thing that makes surveillance very beneficial is the ability to halt so many hazards with the help of surveillance rounds. Emergency situations that call for an immediate response might range from life threatening, disaster struck or menacing situations. Security robots can report incidents to human supervisors or initiate the right security measures that can help to minimize the impact through mobility and instantaneous reporting. It has been observed that robotic security and surveillance systems may provide a feasible solution to enhance security systems in workplaces factories or any other business areas.[10]

2.4) SURVEILLANCE ROBOT TECHNOLOGIES

2.4.1) SENSORS AND PERCEPTION

2.4.1.1) Camera sensing and image detection:

To handle the many monitoring capabilities of the system, the surveillance system utilizes the Raspberry Pi as the controlling unit as well as an added module known as the camera pi to supply the input image. This application enables the Raspberry Pi to switch on the Pi-Camera and capture an image at the onset of a video stream. The video's image is first of all identified as a "face", then cropping and featured extraction is done with the help of the HAAR formula. With the Wi-Fi adapter, he send the system the processing image. However, before this is done, the CNN method for face recognition is brought in to analyze the detected human face against the other faces that have been recorded in the database.[5]

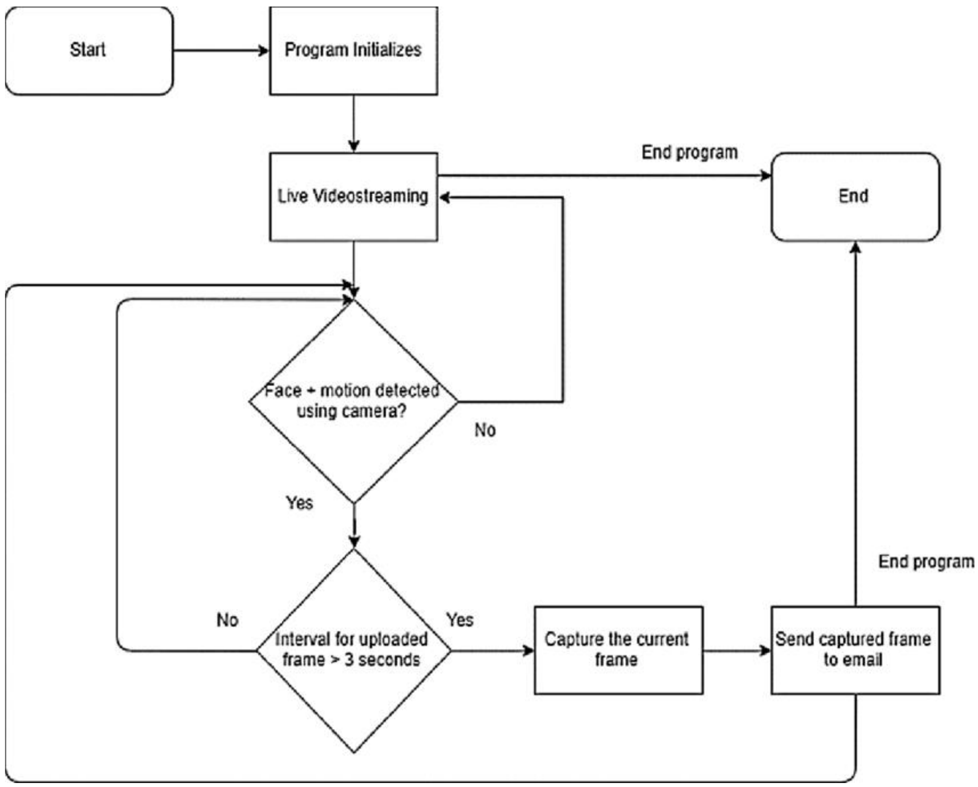

Figure 2.4 Flow chart of developed system using camera with OpenCV implementation

2.4.1.2) **LDR and Ultrasonic Sensors:**

The LDR sensor is used to measure low light intensity that exists in the environment. It means the robot is endowed with the capacity of being able to detect if it is dark, and as a result of this it will alert the homeowner by switching on the lights closest to it. Final results of the experiment with the robot and specifically its trial results are also part of the publication. These results also showed that through programming; the robot was capable of perceiving movements and notifying the homeowner. When there was blurring of light during the day, it could also automatically switch off the lights during the night. Lastly, the outcomes presented shown how effectively the robot operated to provide the house with another layer of security.[11]

Using DS ultrasonic sensor to avoid an obstacl When detecting an obstacle through the condition distance of the sensor, which is less than 60 cm K through the ultrasonic sensor and turns right as it starts moving in that direction. If it is determined by condition distance left, it turns left, and from that moment, it continues to move in that direction if an obstacle is in the way. In this case, the ultrasonic sensor reverses the movement of the robot depending with the condition of right, left ,and all the sides, if the obstacle is detected through a condition distance of (> 60 cm). Each time a stranger or an object moves within a detection line of the PIR sensor, it triggers an alarm. Here the PIR Sensor detects the motion of the object and sends the information to the Arduino Board. [5]

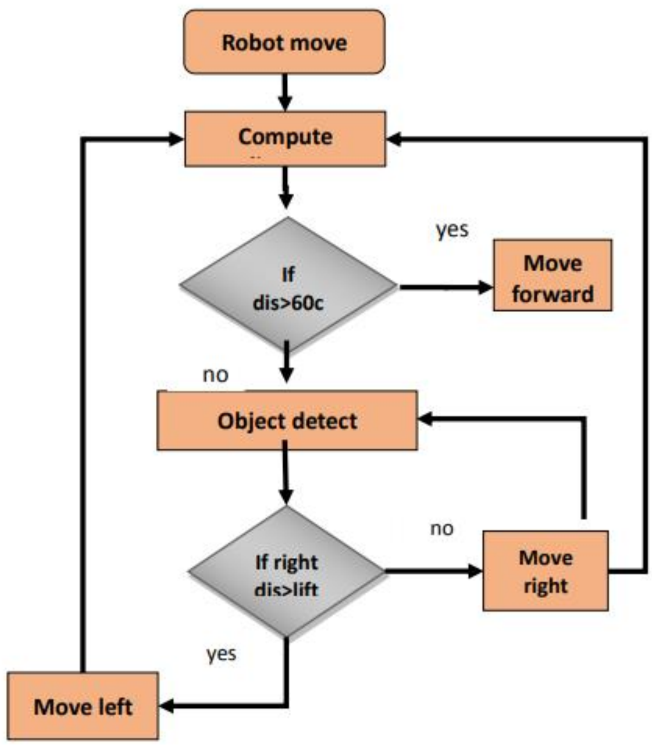

Figure 2.5 Flowchart for Robot Movement

2.5 MECHANICAL DESIGN ANALYSIS

2.5.1) ANSYS

Numerical tool suites frequently are used as a highly articulated series of tool known as ANSYS software that provides a comprehensive means of analyzing systems and modeling their degrees of response to a wide range of environments. Indeed it has become a fundamental design tool for engineers and designers as it provides physical simulation allowing them to predict the behavior of structures, systems, parts under various loads and environmental conditions without having to produce a prototype. The ANSYS software suite contains numerous programs that apply to almost all fields of engineering. They include electromagnetic design as it normally involves the analysis of electromagnetic fields, circuits, antennas, and devices; structural design, which involves stress and strain, deformation, and stability of structures; thermal design, which involves the prediction of temperature distribution and the thermal effects in components; fluid and heat as it involves fluid flow and heat transfer; and acoustics as it concerns sound and vibration attributes of structures and components.

The objective of this paper is to present a brief overview of ANSYS, which is at its core a numerical method of Finite Element Analysis FEA that employs approximations to predict the behaviour of a physical system. Through FEA, large structures are solved into manageable smaller structural elements making it possible to study numerous aspects in detail including stress, strain, and heat behavior. All important steps are worked out and described in detail within the scope of the software. Ansys works in the same way, however it involves preparation of the geometry, importing CAD models or generating geometries within ANSYS before pre-processing and meshing and finally meshing, which ensures a correct discretization of the issue at hand. What we get here is that the solver performs complicated calculations for modeling the physical relations and different values such as temperature, tension, and strain. Post-processing in ANSYS involves depicting simulation outcomes as trends, plots and animation thus it is crucial. People can obtain important information in order to study that information or compare it to the results of experiments offered by engineers.

A solid understanding of the material's mechanical, thermal, electromagnetic, and hydrodynamic characteristics is critical for simulation and is well explained and modeled in ANSYS.

Also, for optimization, there is easier means to get to the design that best satisfies a set goal within set constraints according to ANSYS. For the purpose of identifying the effects that parameters have on a system, the process of choosing parameters that would have an impact can be done systematically, through a process known as parametric analysis. It also integrates nicely with CAD software, thus, the geometry and models can be easily transferred along with models and with programming languages like Python, for customization and automation of the simulation. Companies rely on ANSYS to improve efficiency and profitability through accurate simulation of designs, minimized costs, and reduced time to market for their products. It is used mostly in aerospace, automobiles, electronics and electric, manufacturing, energy, and health care sector. Achieving direct, precise estimation greatly enhances the reliability and efficiency of engineered systems and products and their ability to adapt to different conditions. Owing to its versatile uses and compatibility, ANSYS remains an indispensable tool in the engineering world as a tool that provides critical input to foster the growth of numerous industries and come up with sound decisions.

2.5.2) **Structural Analysis:**

The first step in the structural analysis involving ANSYS is simply to integrate the 3D geometric model of the robot as a part of the EDT and in the context of a previously developed robot design in SolidWorks. In the context of this study, careful selection of components and materials by creating models, connections and other structural parameters in the ANSYS environment is appropriate. The next important step is to provide each of the parts of the robot with certain properties associated with the material. When launching the model in ANSYS, the material properties chosen in SolidWorks are employed to predict accurate simulation results based on such fundamental parameters as Young's modulus, Poisson ration, density, and possible nonlinear parameters.

After being imported the model gets mesh generated with the required material attributes used in it. Since the geometry of the components can easily be distorted and annotated in a finite element model, ANSYS sub-divides the model into finite elements for maintaining the geometric accuracy of the design. The mesh quality and density play a major role in determining the reliability and feasibility of the subsequent simulation results. Following the generation of the mesh geometric quantities are employed to define the robot's behavior at the boundary of the computational domain by defining the forces exerted as well as constraints of certain DOFs at specific points. These are then applied on the SolidWorks design of the robot in the expected loads and forces that the robot will encounter while in function. The particular analysis type is then employed depending on what has been chosen, which could be stress and deformation analysis, modal analysis for frequency or dynamic stress analysis for time varying loads. In light of the input in terms of the specifications of the used material, loads that are to be assumed, and the boundary conditions, the ANSYS solver provides solutions as to the distribution of stresses, deformation as well other mechanical parameters.

The post-simulation analysis in ANSYS is way more comprehensive as it provides charts, graphs and different visualization tools to analyze the result. From the structural mechanics perspective, engineers might obtain knowledge about the performance of the robot, as well as the stress distribution and deformation of the structure as a result of loading in terms of SolidWorks.

2.6 FACE RECOGNITION

Face recognition is a technique that uses a person's facial traits to recognize or confirm their identification. It is a biometric technology that compares and analyses facial patterns using a variety of algorithms and mathematical modelling.

2.6.1) **Methodology**

The biometric technique of face recognition involves matching or verifying the link which is associated with a face that has previously been identified in a frame of an image or a video. Many of the systems then go for alignment and normalization that help in placing and resizing the face for further analysis uniformly in the next stages. It then scans and measures prominent facial features and quantifies them into references that are easier for the algorithms to process – the feature vectors. Then these feature vectors are stored in the database and are often linked to certain individuals. To implement the capability of face recognition, the facial feature vector is first obtained from the target face and then matched with the stored facial feature vectors in the database. The match between these vectors, and in essence the identification of detected face, in identified faces is dependent on their similarity. Biometrics which involve the use of faces has applicability a number of affordances, such as identity confirmation, inclusion systems, monitoring, and other correlatives. Concerning privacy and ethical points, the utilization of face recognition, especially in public or conspicuous circumstances, must be approached with great caution.

2.6.2**) Evolution of Face Recognition**

Although the field of face recognition was introduced in the 1960s, its understanding began with rudimentary approaches by scholars Bledsoe among others. As an extension of it, Bledsoe introduced the system that involved linear dimensions of facial features along with geometric representation.

Eigenfaces was an innovative concept back in the 1970s by Matthew Turk and Alex Pentland where the linear characteristics of face were defined as eigenvectors using the PCA method. This method was one of the essential developments because it extended the possibilities for reducing facial features proportions, as well as formed the basis for the modern face recognition programs. Eigenfaces is one of the most often implemented features of PCA and is used fore preprocessing facial image data. They help to decrease dimensionality while retaining important features that are useful in preparing the data for the following recognition processes. In other words, in face recognition, when a decision has to be made on how similar a new face is to the other faces then, projections of a new face in eigenspace can help identify the people.

The first facial recognition system was developed by Christoph von der Malsburg and his colleagues and by late 1980s there was commercial facial recognition system. Pioneering the face recognition technology for its communicability utilizations and establishing the foundation for the useful application, this system focused on regional facial characteristics.Eigenfaces, emerging in the late 1980s and early 1990, are based on PCA, known to have were first introduced. This method was designed by Matthew Turk and Alex Pentland; it qualifies the facial images to be plotted in two-dimensional space. The facial traits and the changes that were to have been reflected by Eigenfaces were the most important.

Facial efficiency has developed at an improved rate in the 1990s in terms of algorithms. Studied literature categorized several methods, such as feature-based and framework-based approach. Concretely, from the LBP invention, Ojala et al. began to show its efficiency to analyse the texture and to recognise the face in 1994. LBP works in a way that compute the relationship of each pixel to other pixel in the image. This is due to the ability to extract textural information from facial photos and as such it is not vulnerable to changes in illumination and other related factors. LBP is one of the most relevant approaches for facial analysis as the features obtained through it are used to sort and recognize faces.

Timo Ahonen, Abdenour Hadid, Matti Pietikäinen from Infotech Oulu's Machine Vision Group have described in their study how LBP is applied in face identification. They detail how the methodology was accomplished in its entirety encompassing the constraints and assessment metrics and the benefits and performance achieved by implementing LBP. The paper presents, therefore, a new technique for face recognition which takes into consideration both geometrical and photometrical aspects of face images. In the article, the facial area is segmented into smaller sub-regions, thus constructing the histograms of the LBP from each one. The face image is therefore represented by hist2 which is a feature vector made up of the concatenation of hist1 as shown below: To support face recognition in the feature space the authors apply a closest neighbour classifier which measures the distance in the Chi square manner. A weighting formula is also applied into the system; this assigns a certain weight to other areas of the face that holds many greatest facial features like the eyes. To evaluate the performance of the proposed method, FERET dataset is used in this work, there are several methods involved for comparison including Principal Component Analysis (PCA), Bayesian Intra/extra personal Classifier, and Elastic Bunch Graph Matching.

It is also found that based upon the comparative analysis of the results of different tests of expression, lighting and fluctuations in ageing, the suggested approach is more efficient than other approaches. But for all the recommendations made in the paper, this last proposed solution is quite simple, efficient and robust against the noises and fluctuating light intensities. The work offers numerous recommendations for method enlargements as well as variations.

The pioneering algorithm that formed the foundation of real-time face detection was called Viola-Jones algorithm, which was made developed in the early 2000s using Haar-like features and a form of machine learning called cascade classifiers. At the same time, modern 3D face recognition was growing rapidly due to the fact that it provided additional depth information to recognize a face more accurately. Still, the development of CNNs and generally deep learning paradigm shift in the mid 2010s brought a new perspective to face recognition. The system then automatically learns similar features without human intervention from the facial images and gradually derives sophisticated patterns that define facial attributes. In face recognition scenario, the trained CNN's are mandatory as they are capable of distinguishing between different people through comparison of learnt features a process that enhances identification efficiency.

Google Brain introduced TensorFlow in 2015 as a fast and flexible machine learning platform. This led to its quickly becoming very popular because of the fact that it was useful in both large and small applications and also full support it gave to deep learning models. CNNs the model that was used in face recognition is one of the deep learning models that have benefitted from the formation of TensorFlow. It also serves as a training benchmark for CNNs and other related deep learning models that could easily help to obtain high levels of facial recognition accuracy. TensorFlow is valuable in this context because it allows for easily modifying it, combining with other modules, and building one-of-a-kind models.

TensorFlow along with CNNs is now widely preferred and applied by investigators for designing accurate facial recognition software. CNN-based model which exhibited the state of art facial recognition rate in a research completed by Liping Yuan1, Zhiyi Qu1, Yufeng Zhao1, Hongshuai Zhang1 and Qing Nian2. TensorFlow's integration showed the capabilities of deep learning for the spot facial recognition.

About the face recognition method suggested in the research, it is found to use deep learning in order to extract information from existing photos of faces. The facial recognition model is trained and tested using Convolutional Neural Networks, also known as CNN and TensorFlow is the basic framework. To evaluate the efficiency of the suggested method, it is worked on the ORL and Yale face datasets and its result is compared with other methods Eigenface, Fisherface, and LBPH. The results therefore show that with respect to proactive strategy the suggested strategy is best suited under complex conditions involving stance, expression, illumination and occlusion.

The article by Houda Meddeb, Zouhaira Abdellaoui and Firas Houaidi entitle "The environmental impacts of energy consumption: Does sustainability matter" is as fascinating as it is informative. regardless of an overview of the surveillance robot that can be created by employing Raspberry Pi and IoT. It refers to locking doors and use facial recognition to detect people who are not supposed to be inside the compound. For motion sensing, the robot is equipped with a PIR sensor and as for the imaging purpose, it is installed with a camera. Such aspects include the fact that the images are analyzed employing facial recognition software in order to recognize a person. This indicates that in case an intruder is identified in the premises being monitored by the robot, it proceeds to alert its owner. The manipulation of the robot is done via a web interface, They used this approach to do away with the complexity of hardware controllers.

At the moment, research areas such as deep learning, multimodal biometrics, and privacy enhancement are pushing the development of facial recognition systems. Like Biometrics, RFID is employed in security and surveillance, access control, payments, retail, and other areas, it is a modern-day technology enabler. Among the essential components of ASRs, face recognition can be noted, which enables the used robot to identify people in a certain area. Great improvement has however been registered in the advancement of this field of technology particularly with the aid of OpenCV and other methods.

2.6.3) Implementation:

After, having identified a face in an image or video frame, face recognition, mostly encompasses the process following that line to authenticate or recognise the person in the frame. The last stage of face detection application is usually followed by face alignment and normalization to guarantee that the face being analyzed is in a constant position and scale. Next, the application outlines facial landmarks related to the object of interest, translating these landmarks into numerical values or vectors. These feature vectors are eventually archived into a database, which is frequently linked to particular users. It extracted face feature vectors from the detected face, when it needs face recognition and matches these vectors with the face vectors it has stored in the face database. The degree of the match can be decided by comparing these vectors and thereon the identity of the face detected is achieved. Consequently, the usage of face recognition can be rather diverse and include identification of the device owner, monitoring people's activities, security systems, identification of the person, and much more. Nevertheless, some concerns regarding privacy and ethical issues may arise when it comes to using face recognition, for instance, in handling large audiences or cases where requested data may be highly confidential.

2.7 IMAGE PROCESSING

2.7.1) Tools used for image processing:

Considering primary deliverables such as object detection and recognition, tools such as OpenCV, media pipe, and yolo have been considered. Pre-trained data sets for object detection are available for variety of different objects. A brief overview of all three is given below.

2.7.2) Introduction to OpenCV:

OpenCV, an open-source computer vision library, has played a pivotal role in advancing the field of computer vision and image processing. It was initially developed by Intel in the late 1990s and later became an open-source project, fostering a global community of developers and researchers. [12]

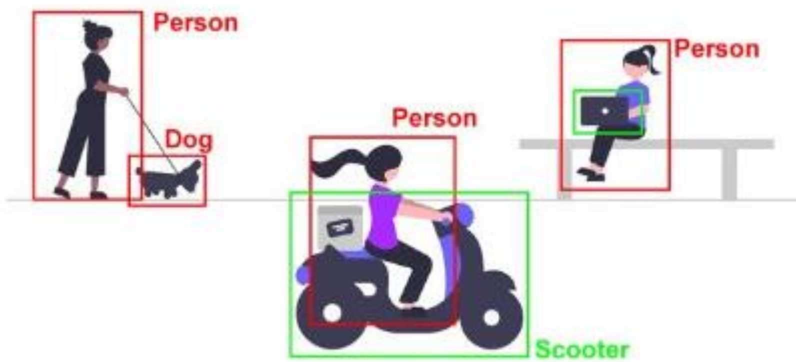

Figure 2.6 Object Detection by Open CV

2.7.2.1) Historical Development:

The history of OpenCV can be traced back to 1999 when Intel initiated the project. Over the years, it has undergone several iterations and improvements. Significant milestones include the release of OpenCV 2.0 in 2009, which introduced a more user-friendly C++ API, and OpenCV 3.0 in 2015, featuring improved support for C++11 and more advanced computer vision algorithms.

2.7.2.2) Features and Capabilities:

OpenCV provides a vast array of tools and functionalities for computer vision and image processing tasks. It offers support for image and video analysis, feature detection, object recognition, machine learning, and more. Its comprehensive documentation, tutorials, and community support make it accessible to both novice and experienced developers. Given below are some of its applications:

2.7.2.3) Applications:

OpenCV finds applications in diverse fields, including robotics, augmented reality, medical imaging, surveillance, and automotive industries. Its ability to perform real-time image processing has made it a popular choice for developing vision-based systems.

1) **Robotics**: OpenCV is extensively used for robot perception, enabling robots to navigate environments, identify objects, and perform tasks like object manipulation.
2) **Augmented Reality:** It plays a crucial role in marker-based and marker less AR applications, enabling the overlay of digital information onto the real world.
3) **Medical Imaging**: OpenCV assists in medical image analysis, aiding in disease diagnosis, surgical planning, and the development of assistive technologies for healthcare.
4) **Surveillance**: OpenCV is an integral component of surveillance systems, facilitating tasks like face recognition, object tracking, and anomaly detection.
5) **Automotive:** It contributes to various aspects of autonomous vehicles, including lane detection, pedestrian detection, and obstacle avoidance.

2.7.2.4) Contributions to Computer Vision Research:

OpenCV has been instrumental in the development and implementation of computer vision algorithms and techniques. Researchers often use OpenCV as a tool to prototype and test their ideas. The library's inclusion of state-of-the-art algorithms makes it a valuable resource for the computer vision community.

2.7.2.5) Challenges and Future Directions:

While OpenCV has made significant advancements, there are ongoing challenges. Keeping up with the latest developments in computer vision, providing support for emerging hardware platforms, and improving compatibility with deep learning frameworks are areas where OpenCV continues to evolve.

Advancements in OpenCV:

1) **Deep Learning Integration:** OpenCV integrated deep learning capabilities, allowing users to easily work with deep neural networks for tasks like object detection, image segmentation, and image classification. It provides support for popular deep learning frameworks like TensorFlow and PyTorch.

2) **DNN Module:** OpenCV introduced the DNN (Deep Neural Networks) module, which enabled efficient inference of pre-trained deep learning models. This module is especially important for real-time applications and edge computing.
3) **GPU Acceleration:** OpenCV leveraged GPU acceleration to speed up various computer vision tasks, making it possible to process images and video streams in real-time or near real-time on compatible hardware.
4) **OpenCV.js:** OpenCV expanded its reach to web applications by introducing OpenCV.js, which is a JavaScript binding for OpenCV. This allows developers to use OpenCV in web-based projects, making computer vision accessible in web applications.
5) **Support for 3D Vision:** OpenCV extended its capabilities to handle 3D computer vision tasks, including 3D reconstruction, stereo vision, and depth sensing.
6) **Real-time Tracking and Augmented Reality**: OpenCV added features for real-time object tracking and augmented reality applications, making it suitable for a wide range of interactive and immersive experiences.
7) **Machine Learning Enhancements:** OpenCV integrated machine learning algorithms and tools, enhancing its capabilities in areas like feature selection, classification, and regression.

2.7.2.6) OpenCV role in Project:

It's important to note that while OpenCV can be used for feature extraction and object detection, the performance and accuracy of this approach may be limited compared to deep learning-based object detection models like YOLO or Faster R-CNN. Deep learning models have proven to be highly effective for object detection and classification tasks and often outperform traditional computer vision methods. As we are detecting and classifying objects in real-time, the computational power available for such an operation much be evaluated. Using YOLOv8 with Raspberry pi 4 we do get accurate results but at a cost of Very low (unusable) FPS such as 1 or even 0.5 fps. Therefore, to get desired results at reasonable Fps we will tweak OpenCV to the optimum settings.

2.7.2.7) How to get better fps in real-time processing?

You can make several optimizations to improve real-time performance and achieve better frames per second (fps) when using OpenCV for computer vision tasks on resource-constrained devices or in applications where speed is crucial. Here are some strategies to tweak OpenCV for better real-time fps:

1) **Lower Image Resolution:** Reduce the input image or video frame resolution. Smaller frames require less processing and lead to higher fps.
2) **ROI (Region of Interest):** If you know the region where objects of interest are located, you can extract and process only that part of the image, saving computation time.
3) **Hardware Acceleration:** Use hardware acceleration when available. OpenCV supports hardware acceleration with platforms like Intel's Open VINO toolkit, which can significantly boost performance.
4) **Multithreading:** Implement multithreading to parallelize tasks. You can use multiple threads to process frames concurrently, improving performance on multi-core processors.
5) **Algorithm Selection:** Choose the right algorithm for your task. OpenCV offers various algorithms for the same task, some of which may be more optimized than others.
6) **Model Quantization:** If you're using deep learning models, consider quantization. Quantized models use lower precision, reducing memory and computation requirements while maintaining reasonable accuracy.
7) **Streaming Buffer:** Implement a buffer for video streams. Process multiple frames in parallel to reduce latency and improve fps.
8) **GPU Usage:** If you have a compatible GPU, utilize it for certain OpenCV operations. OpenCV can offload some tasks to the GPU for faster processing.
9) **Memory Management:** Optimize memory usage by reusing data structures and minimizing memory allocations and deallocations.
10) **Algorithm Subsampling:** For some applications, you can skip processing some frames or subsample frames to reduce the computational load while still achieving acceptable real-time performance.

Keep in mind that the effectiveness of these optimizations depends on the specific computer vision task and the hardware you're using. The right combination of these strategies can significantly enhance realtime fps while maintaining the required level of accuracy for your application.

2.7.3) Media pipe introduction:

Media Pipe is a framework developed by Google for building machine learning-based applications and solutions for various media-related tasks, including but not limited to computer vision, gesture recognition, facial recognition, and pose estimation. t provides a wide range of pre-built models and tools to facilitate the development of applications in areas such as:

1) **Gesture Recognition:**

   Media Pipe can recognize and track hand gestures, enabling natural and intuitive interaction with devices and applications. It's commonly used for gesture-based control systems, sign language recognition, and more.

2) **Facial Recognition and Tracking**:

   Media Pipe offers facial recognition and tracking capabilities, allowing developers to detect and track faces in images and video streams. This is used in applications like face authentication, emotion recognition, and augmented reality (AR) face filters.

3) **Pose Estimation:**

   Media Pipe can estimate the pose of a person, including key points representing joints and body parts. This is valuable for applications like fitness tracking, sports analysis, and gesture-based interfaces.

4) **Object Detection and Tracking**:

   It supports object detection and tracking in real-time, making it useful for applications like surveillance, robotics, and augmented reality.

5) **Hand Tracking:**

It offers precise hand tracking, which is essential for applications like virtual reality (VR) controllers, sign language translation, and hand gesture recognition.

6) **Pose Estimation for Animals:**

Media Pipe has been used to estimate the poses of animals, such as dogs and cats, in research and pet-related applications.

2.7.3.1) Advancements in media pipe:

1) **Additional Pose Models:** Media Pipe expanded its suite of pose estimation models to cover various use cases. This included models for full-body pose estimation, multi-person pose estimation, and even custom pose estimation models.
2) **Media Pipe on the Web:** Google introduced Media Pipe for the web, enabling developers to leverage the framework in browser-based applications. This opened up opportunities for creating web-based computer vision and augmented reality (AR) experiences.
3) **Face Detection and Mesh Models:** Media Pipe introduced models for face detection and facial landmark detection, including 3D face mesh estimation. This allowed for applications such as facial tracking, expression analysis, and augmented reality face filters.
4) **Holistic Models:** Media Pipe introduced holistic models that combine pose, face, and hand tracking in a single model. This offered a more comprehensive understanding of the user's body and hands in applications like fitness tracking and immersive AR/VR experiences.

2.7.3.2) Media pipe Role in project:

We will utilize this tool to locate the human face and for human detection. Using media pipe we can get key points of the human body, which mainly consists of joints. This allows us to adjust the FOV for better image processing operations on the face. Since our main concern is dealing with humans and identifying potential outliers this algorithm is perfect as it can detect anomalies such as various gestures and can track poses, which contributes to the process of making key points of human body.

2.7.4) YOLO (You Only Look Once)

2.7.4.1) Introduction to yolo:

The YOLO (You Only Look Once) object detection algorithm represents a significant advancement in computer vision, offering a highly efficient and accurate solution for real-time object detection tasks. Developed by Joseph Redmon and Santosh Divvala, YOLO has become one of the most popular and influential algorithms in the field of computer vision.

2.7.4.2) Historical Development:

YOLO was first introduced in 2016 with the release of YOLOv1, which pioneered the concept of one-shot object detection, meaning it could detect and classify objects in a single forward pass of a neural network. This approach marked a departure from traditional object detection methods, which typically involved multiple stages.

2.7.4.3) Key Concepts and Advantages:

1) **Single Pass Detection:**

   YOLO's primary innovation is its ability to perform object detection and classification in a single pass, making it extremely fast and suitable for real-time applications.

2) **High Accuracy:**

   Despite its speed, YOLO has demonstrated impressive accuracy in detecting a wide range of objects, thanks to its deep neural network architecture and large-scale training on diverse datasets.

3) **Anchor Boxes:**

YOLO uses anchor boxes to improve object localization. These anchor boxes help the model predict bounding box coordinates more accurately.

2.7.4.4) Applications of yolo:

Some of the key operations and applications where YOLO can be effectively used include:

1) **Object Detection and counting:** The primary purpose of YOLO is real-time object detection. It can identify and locate objects of interest in images or video frames, providing bounding boxes around detected objects along with class labels and confidence scores.

2) **Single-Class and Multi-Class Detection:** YOLO can be used for single-class object detection, where it identifies instances of a specific object class (e.g., detecting only cars). It can also perform multiclass detection, where it detects and classifies multiple object classes simultaneously (e.g., detecting cars, pedestrians, and bicycles in the same image).

3) **Real-Time Video Analysis:** YOLO is well-suited for real-time video analysis and tracking. It can track objects across consecutive frames in video streams, making it valuable for surveillance, traffic monitoring, and action recognition.

4) **Object Recognition and Classification:** Beyond object detection, YOLO can classify detected objects into specific categories or classes. For example, it can not only detect cars but also classify them as sedans, SUVs, or trucks.

5) **Face Detection and Recognition:** YOLO can be adapted for face detection and recognition tasks. It can locate faces in images or video frames and associate them with identities.

6) **Augmented Reality (AR):** YOLO can be employed in AR applications to detect and track objects or markers in the real world, allowing for the overlay of digital content onto the physical environment.

7) **Custom Object Detection:** YOLO can be customized to detect specific objects or classes relevant to specialized applications, such as detecting specific components in manufacturing or items in a retail setting.

2.8 COMMUNICATION AND DATA TRANSMISSION:

The general field of manufacturing, housekeeping and other popular entertainment areas involving robots and automation systems can be said to be one of the simplest fields across the globe. A project proposed to design an automatic car that can be controlled through an android device to include a wireless camera for surveillance purposes. One of the features of this robot is that it can stream live video with low illumination and this is very essential because there is always a need for surveillance during a war. Wifi technology is comparatively a new technology though it has the potentiality for growth and can become practically usable in the near future. Mobile device has a built-in application referred to as the android application which in turn relays signals to a security system that processes commands. The motion detector and the CMOS camera are worn for remote surveillance are worn. Robots is defined as a mechanical device that is used for performing certain operations in a routine and efficient manner and that is controlled by a computer. The type can be controlled from a distance by a human being, in which case wireless communication is vital. That is like having a human guard to monitor the security aspect; the robot offers security by patrolling in an automatic vehicle at specified or a limited region.[13]

2.8.1) Wireless networks and data transfer

In different locations, there is always a need to constantly monitor something. Modern surveillance systems include CCTV cameras and other monitoring systems to take control anywhere. These systems mainly may provide a limited coverage region and are therefore not portable at all. These are mainly operated by hand or with the help of computers. They cannot be adapted to serve a vast area and similarly, they can be operated through any portable device. Lastly, based on the points made above we can conclude that these systems are not dynamic. that one had to develop a far more extensive system of watching the world. Dynamic and remote controllable. This is the reason why the purpose of the given project will be the establishment of a surveillance system that can be controlled via Android application. At the center there is a sitting which is like a seat which has a robot and there is a wireless camera attached on it. This technical equipment relays the high definition video stream back to the affiliated Android device that is used to control the robot.[13]

2.8.2) Integration with the Internet of Things (IoT):

User end of the robot can be carried laptop or mobile for interacting with the robot. Thus the user section may involve a laptop or a mobile can make the said section a portable one contrary to those that require a general stationary computer system. It can be done by RF technology or by Zigbee device or through Bluetooth technology but that little short comes with loss of range. Therefore in an effort to bring in the targeted idea of raising the range we can try linking the user section with the internet which forms the essence of what is known as the internet of things. For the purpose of integrating the introduced user system with the internet, it is possible to utilize the BLINK/CAYENNE software. CAYENNE software is not more than an object relational mapping (ORM), particularly used to define the prototypes and development of the IOT applications. Therefore it is very important that through this CAYENNE software, we can send commands and can easily control the Robotic vehicle. While, Blink software may denote ranges of software including but not limited to Blink Charging Software, Eye-Tracking Software, Website and Application Testing Software, Rapid Application Development Tool and, Instant Messaging Software. This software can be used in many ways, including the charging of EVs, analysing the website or app performance, creating simple apps, sending messages and communicating with others.[14]

2.8.3) Key features:

The app is developed for Android and iOS platforms. Key features include

User Authentication: To ensure security, users may need to log in before accessing the robot.

Live Video Streaming: Real-time video feeds from the robot's camera are displayed within the app.

Remote Control: Buttons in the app allow users to control the robot's movement.

Sensor Data Display: Sensor data, such as distance measurements from ultrasonic sensors, is presented in a user-friendly format.

Alerts and Notifications: Users receive alerts when the robot encounters obstacles or when specific conditions are met.

Communication with Raspberry Pi: The app communicates with the Raspberry Pi over Wi-Fi using a predefined protocol.

2.9 CHALLENGES AND LIMITATIONS

Creating a surveillance robot involves various technical and logistical challenges. Here are some of the key challenges you may encounter:

Navigation and Obstacle Avoidance: Developing a system that allows the robot to navigate autonomously through complex environments while avoiding obstacles is a significant challenge. This requires robust sensor systems (e.g., cameras, LiDAR, ultrasonic sensors) and advanced algorithms for path planning and obstacle detection.

Battery Life: Surveillance robots often need to operate for extended periods without human intervention. Maximizing battery life is crucial, and efficient power management and recharging systems are required.

Communication: Maintaining a reliable and secure communication link between the robot and the control center is vital. This includes handling potential issues like signal interference, range limitations, and encryption to prevent unauthorized access.

Environmental Adaptability: Surveillance robots may need to operate in various weather conditions and terrains, so they must be designed to withstand and perform in adverse situations.

Data Processing and Storage: The robot's sensors generate a substantial amount of data, such as images and videos. Efficient data processing and storage solutions are essential for real-time analysis and historical data retrieval.

Security and Privacy: Implementing strong security measures to protect the robot from hacking or tampering is crucial, especially when the robot is used for surveillance purposes. Privacy concerns also arise when collecting and storing data in public or private spaces.

Human Interaction: Ensuring that the robot can interact with humans in a friendly and understandable manner is important. This includes user interfaces for remote control and effective communication methods.

Maintenance and Reliability: Surveillance robots require regular maintenance to ensure they remain operational. Developing a design that is both reliable and easy to maintain is essential.

Cost: Building an effective surveillance robot can be expensive due to the advanced sensors, hardware, and software required. Finding a balance between functionality and cost-effectiveness can be challenging.

Integration with Existing Systems: In many cases, surveillance robots need to work in conjunction with existing security or surveillance systems. Ensuring seamless integration with these systems can be complex.

Ethical and Social Acceptance: The deployment of surveillance robots can raise ethical and social concerns, such as issues related to privacy invasion, surveillance overreach, and public acceptance. Addressing these concerns is important for successful adoption.[13]

CHAPTER 3: METHODOLOGY

3.1 Sensor Fusion Frame work

3.1.1Overview:

As part of this project a sensor fusion framework is developed that uses both ultrasonic sensors in combination with information from stereo cameras in order to expand the robot's perception and movement capabilities. These sensors provide input data into the framework in a mutually enhancing fashion such that the robot can collect a reliable and accurate mobile environmental data that is analysed in real-time.

Ultrasonic sensors are used to sense the radar quality which indicates the distance from the obstacle so that the robot could make sure of surroundings within close range. These sensors generate the ultrasonic waves and exactly calculate the time it will take for the wave to bounce back to know the distance accurately. To solve this, there must be multiple ultrasonic sensors placed around the robot to achieve a complete 360-degree Field of View that provides substantial coverage or high obstacle detection rate.

At the same time, optical sensors are employed to extract three-dimensional information from the environment using stereo cameras. Comparing the differences of images that the left and right cameras record, the stereo vision system will create depth data for the robot to shape a three dimensional environment. This will also enhance object detection and recognition, from objects that could be obstacles, human beings, and other objects of interest definitely within the environment.

Sensor fusion architecture combines the information from ultrasonic sensors and stereo cameras at the cost of high computational complexity with approaches like Kalman filtering, Bayesian learning, or deep neural networks. Combining information from more than one sensor gives the framework better reliability and robustness in perception to match the limitations and uncertainties of a particular sensor measurement. However, the crucial advantage of this approach is that it provides the robot with the ability to predict its location and make better decisions when navigating dynamic environments and tricky real-world environments.

In general, the use of sensor fusion is central to the robot's perception and navigation capabilities and is important in ensuring the robot's safe, efficient, and autonomous navigation and traversal of the environment. As a result of the framework's ability to facilitate the fusion of ultrasonic sensor signals and stereo camera signals, the robot becomes capable of efficiently tracing its movements, which is important for the realization of its surveillance goals as well as for the safety and security of the robot's operations.

3.2 Hardware specifications:

3.2.1. Ultrasonic sensor

An ultrasonic sensor is used in many robotics and automation applications as it works on the principle that the sensor emits ultrasonic waves that bounce off the objects in their path and the distance to the object is calculated from the time taken by the wave to reflect off the object and return to the sensor. The sensors have an ultrasonic transducer which converts electrical signals into sound waves and vice versa and measures distance on the basis of time taken for sound waves to return to the transducer. Its detection range could range from a few centimeters up to several meters, and its measurements could be accurate in this range. Having a wide field of view and possessing the ability to operate in different environmental conditions ultrasonic sensors are applicable in defining object's position and providing proximity sensing. They are great at functioning in uniform sound setups but may struggle in rooms with soft materials that will prevent wave reflections. Nonetheless, they are widely used in robotics, industrial automation, automotive systems, and other consumer electronics in providing reliable solutions to distance measurement, obstruction, and liquid level detection. Ultrasonic sensors are the prominent sensing technologies that are of significant importance in the field of application in a broad area of applications.

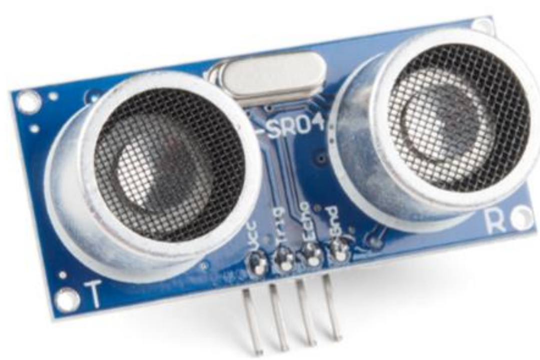

Figure 3.1 Ultrasonic Sensor

3.2.2. Logitech c310 Webcam:

The C310 is an affordable and versatile webcam that supports a variety of applications from live broadcasts, video calls, and production. The C310 is such a small yet efficient and versatile machine for home and office use.

The Logitech C310 is rather easy to use because it is a plug-and-play device and only requires a USB port to function with its built-in microphone that is supported on most operating systems without the need to install the extra drivers. You only need a USB connection to the PC to start recording your audio and video files.

The C310 camera offers video resolutions of up to 720p HD, which makes the camera suitable for video calling and videoconferencing platforms such as Twitch, Zoom, and Skype. The built-in autofocus feature ensures that the images remain sharply focused even in diverse lighting conditions.

Customers may also adjust the settings of the Logitech C310 with Logitech's webcam software such as the brightness, contrast, and color balance with the webcam's operating system. Learners can control their video output depending on ambient lighting and conditions thanks to this versatility.

The Logitech C310 is a great travel computer accessory or a perfect webcam companion for any remote worker or content creator because it is small, lightweight, and extremely portable. The C310 camera is an excellent choice for people who want to quickly set up a camera for streaming games or virtual meetings and vlogging.

Figure 3.2 Logitech c310 Webcam

3.2.3 Rasberrypi-4:

The Raspberry Pi 4 represents the main computer for our robot project which has enhanced a number of important characteristics of the device.

First of all, we have taken advantage of the quad-core ARM Cortex-A72 processor that can achieve a maximum clock frequency of 1. Wi-Fi 5 GHz included in the Raspberry Pi 4. This helps us both in running simple control jobs as well as intensive image processing or machine learning tasks on our robot.

We have also interfaced with sensors, motors, and other peripherals by using the GPIO pins available on Raspberry Pi 4. These GPIO pins mean that through the connection of sensors, our robot is now able to collect environmental data and is able to make decisions in real time based on feedback from its sensors.

In addition, we have integrated a camera in our robot through utilizing the camera interface of the Raspberry Pi 4. This makes it possible for us to take pictures and videos which enable vision based tasks like navigation and object identification as well as detection to be performed.

The robot has inbuilt Bluetooth and Wi-Fi capability that ensures communication on a wireless network. It also means that more sensors and peripherals could be included without being constrained by physical connections, combined with remote control, telemetry, and data transfer.

Moreover, we have explored the expandability of the Raspberry Pi 4 by connecting it to custom hardware through the GPIO pins and to external units through the USB ports. It has been possible to adjust and improve robot capabilities to our project requirements due to the latter's modularity.

On the whole, the Raspberry Pi 4 has been an excellent cost-efficient platform for this robot project due to its powerful computing power, expandable I/O peripherals, and numerous software choices for developing the project.

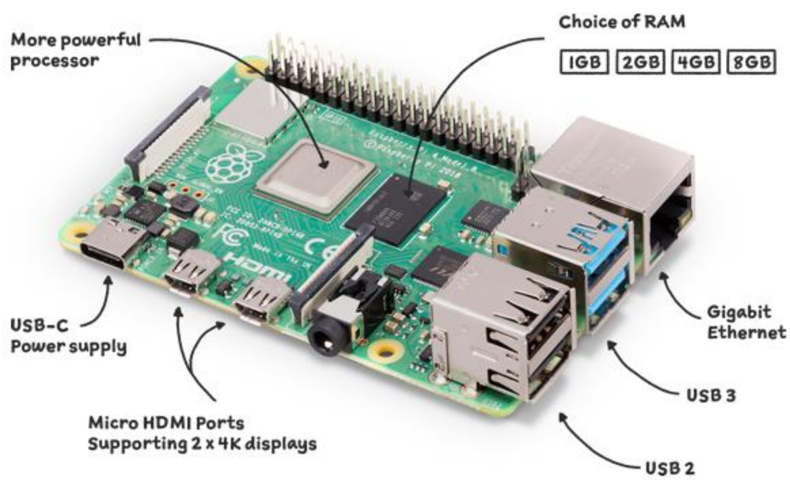

Figure 3.3 RasberryPI-4

3.3 Software:

3.3.1. Open CV:

The OpenCV (Open Source Computer Vision Library) is considered as rock in the field of computer vision as it is offering various tools and algorithms for the image and video processing. This open-source library can perform various tasks including basic image processing, image transformations, feature detection, machine learning, and deep learning. OpenCV is a cross-platform library that facilitates the use of programming tools such as Python, C++, and Java languages to create vision-based applications essential for developers and researchers. It has a wide range of pre-trained models and algorithms to perform computer vision functions such as object identification, facial recognition, object segmentation, and movement analysis. In addition to this, OpenCV has been designed as a modular library that has various integration capabilities with other libraries and frameworks, making it a useful tool for development. OpenCV remains a technology that is deeply integrated at the core of Image Security and Surveillance and Augmented Reality Systems, Digital Medical Imaging Applications and many other areas. Since it is one of the most popular tools in the OpenCV community, OpenCV has continuously led the way in the field of computer vision, open source, and cross-platform vision-based technologies.

3.3.2. YOLO

YOLOv8 Nano is an efficient framework of the YOLO object detection system designed for resource-limited devices such as a device with embedded systems. The primary improvements in YOLO v8 Nano are in the architecture of the neural network and improvements in the speed and performance optimization of real-time object detection which can achieve very high levels of accuracy.
Thus, the YOLO v8 Nano version with an optimized model obtains a middle ground between model complexity and computation speed, which is particularly suitable for applications with severe computing constraints. This model features a one-stage detection pipeline and its compact size ensures that it performs at the speed of a model consisting of 250 million parameters, without sacrificing accuracy.

The design of the YOLO v8 Nano has been achieved by incorporating various optimizations to enhance the inference speed and alleviate memory consumption as much as possible, which makes it an ideal candidate for mobile devices. These optimizations that include effective network architecture, feature reuse and quantization techniques enable it to be highly efficient whereas objects extremely diverse in shape can be detected with a high degree of accuracy. Being small but offering good performance in various environments makes YOLO v8 Nano a viable choice for smart sensor, robotics and surveillance projects. Its average detection time that is reasonably low and efficient computation makes it an ideal solution for resource-limited applications such as edge computing.

3.3.3. Arduino Mega

The Arduino Mega is a powerful microcontroller board used in electronics projects with a significant number of I/O pins. It is an enhanced version of an Arduino Uno that can be used for complex projects because it has more memory, digital pins, and analog pins than the first model. The Mega has many opportunities for peripherals integration because it has 54 digital input/output pins, 16 analog inputs and larger flash memory. It is a favourite choice for designing and developing various electronics among hobbyists, students, and professionals because of its compatibility with many sensors, shields, and libraries and extensive documentation and support from the community. The Arduino Mega is a great device that can help in actualizing simple or complex ideas in robotics, home automation, or data logging fields, among others. Its adaptability and upgradability make it the best option.

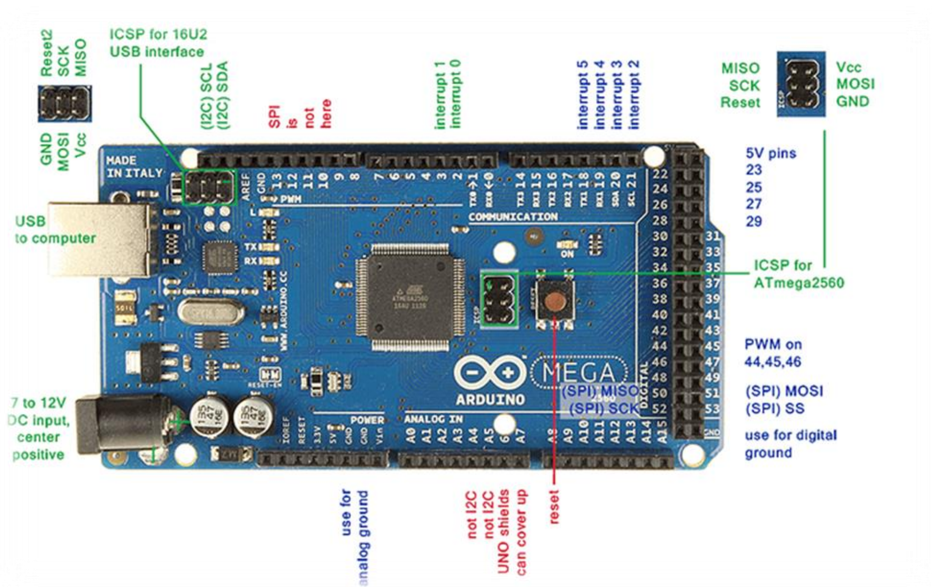

Figure 3.4 Arduino MEGA

3.3.4. Flask

Flask is a micro web application framework written in Python. Flask is considered as a light weight development environment with a flexible architecture. It is a small core that allows for essential web development capabilities like routing with @app. route decorators and content dynamic generation using the Jinja2 templating engine. It is simple and easy to learn and use with the concept of providing an extensible framework that allows the developer to add new extensions to the project using new libraries. This simplicity and flexibility make Flask a suitable framework for small applications and more sophisticated projects while being backed by extensive documentation and an active community.

3.4 Algorithm Generation:

3.4.1. Data Acquisition Using camera:

We mounted the camera module on the chassis of the robot and connected it to the Raspberry Pi 4 using the CSI port to acquire data using the camera on our surveillance robot. We installed and configured the appropriate software: the picamera library for camera access and OpenCV for image processing. The Python scripts set up the current camera settings and captured frames in real-time while saving them temporarily for immediate processing. OpenCV was utilized for image processing to improve the frames and object models like YOLO were used to recognize and categorize objects with bounding boxes. The processed data was used to make real-time decisions such as redirecting the path of the robot to avoid obstacles or illuminating alerts when intruders were present. We retained relevant image data for further analysis and allowed live video streaming for remote monitoring. We managed to achieve this by correlating the data received by the cameras with the ultrasonic sensors and thus created a complete representation of the environment that was sufficient for accomplishing the robot's navigation and surveillance objectives. This integration greatly improved the capabilities of the robot as a real-time surveillance device.

3.4.2. Data Acquisition using Ultrasonic sensors:

Obstacle avoidance was done using ultrasonic sensors in the development of our surveillance robot. This important function allows our robot to recognize and avoid obstacles in its path efficiently and for a sustained period throughout surveillance tasks.

The first step that we took was to connect each of the ultrasonic sensors to the microcontroller board so that every possible angle was covered. Each sensor was positioned in the best way to provide full obstacle detection.

We then set up and configured the pins and established stable connection between the sensors and the microcontroller board throughout the coding process. This arrangement facilitated the sequential triggering of the sensors so that the sensors would be able to produce brief pulses and accurately time how long it takes for the echoes from any surrounding obstruction to return. We soon calculated the distances to the identified barriers using the measured time and a simple formula based on the sound speed. This procedure was repeated for each sensor to give a holistic view of the robot's surrounding and any obstacles that may be present.

With these distance measures our robot can change its trajectory and alter its path in real time to avoid any recognized obstacles. We will ensure quick and manoeuvrable navigation in intricate and constantly changing situations by integrating ultrasonic sensor data into our robot's control system effortlessly.

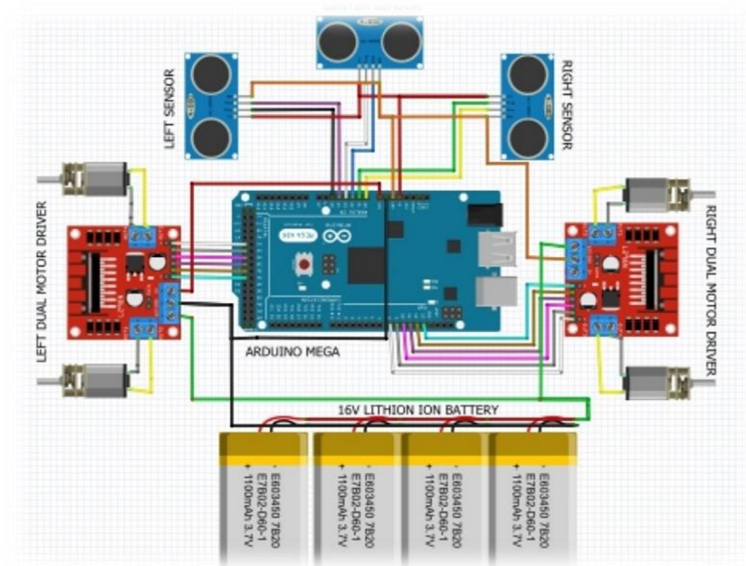

Figure 3.5 Connection of Ultrasonic Sensors and Arduino

3.4.3. Webcam Calibration

In order to calibrate our camera for our surveillance robot we followed an organized method. First, we arranged an 8 x 6 checkerboard pattern and securely mounted the robot's camera. In good daylight we took photos of the checkerboard from different angles and distances.

To locate the checkerboard corners using OpenCV functions, I installed the opencv-python package and wrote a Python script. We obtained 3D object points and matching 2D image points from the recorded photos. We used OpenCV's calibrateCamera function to calculate the distortion coefficients and camera matrix, and we saved the results for later.

So we put the stored calibration data into our main robot program and the robot can real-time undistort the camera's collected images. This was done using OpenCV's getOptimalNewCameraMatrix and undistort functions with the camera matrix and distortion coefficients to ensure accurate and undistorted pictures.

This calibration procedure improved the accuracy of our robot's real-time obstacle detection and movement due to the provision of reliable visual information that our robot needed for its function as a surveillance machine.

3.4.4. Integration of Webcam and Ultrasonic sensors and RaberryPi-4:

In order for our surveillance robot to synchronize the camera, Raspberry Pi 4, and ultrasonic sensors, we needed to use a combination of hardware and software to create the system. I arranged the camera module to capture the scene around the robot and placed the ultrasonic sensors on the body of the robot in such a way that they can reliably detect obstacles. The sensors and camera were connected to the correct wires of the GPIO pins of the Raspberry Pi 4. We developed Python scripts to fetch distance readings from the ultrasonic sensors through the RPi. GPIO module then used logic to determine barriers based on these readings to obtain sensor data. We also used the camera module to simultaneously capture and process photos with OpenCV and develop object detection and recognition algorithms.

We used the combination of the camera and the ultrasonic sensors to process data in order to make the right decisions about the robot's movements. This facilitated integration of sensor inputs to allow dynamic navigation and task performance. The robot changed its direction and speed in real time depending on the processed information from the sensors and the camera using suitable control methods developed for the Raspberry Pi 4. It also made it possible for users to access the camera feed in real-time, detect obstacles, and warn about security hazards or interruptions. This interface improved situational awareness by giving text messages and visual and sound alerts.

We finally implemented a comprehensive testing and optimization procedure to ensure the dependability and performance of the system under different situations. We were able to optimize the parameters and algorithms of the surveillance robot to enhance its accuracy and efficiency.

With the use of this integrated methodology, we have been able to design a reliable surveillance robot that can be used for a number of surveillance applications. It is able to recognize potential obstacles, collect videos, and make decisions.

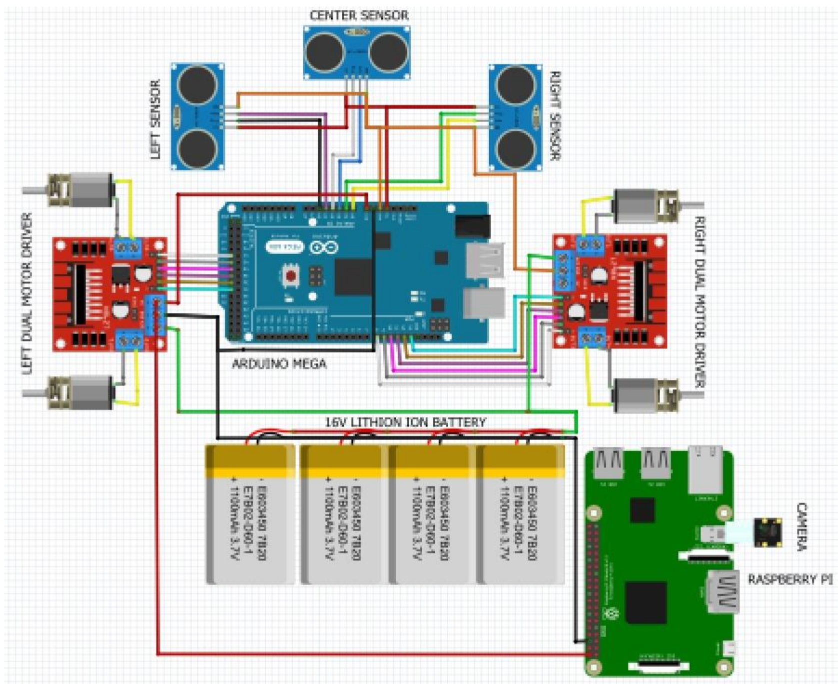

Figure 3.6 Electronics

3.4.5. Utilizing YOLO v8 Nano for Object Detection and Recognition:

a) Dataset Preparation:

Selection of Dataset: Photos in our dataset were pre-labelled and annotated for fifty classes. These classes included a wide range of objects that are used in common places like cars, animals, furniture and electronics.

Data Verification: Before using the labelling, we carried out verification and validation to ensure the accuracy and consistency of the labelling of all samples in the dataset. This procedure was important to ensure that the dataset was not compromised and that the models generated were not biased.

Data Splitting: The dataset was split into two distinctly separate subsets: a training set and a validation set. The training set consisting of 80% of the data was used to train the model while the validation set consisted of the remaining 20% of the data which was used to evaluate the performance and the generalization of the model. This division ensured that the model's performance could be tested out on data that it had not encountered before.

Data Preprocessing: The dataset was pre-processed before training to adhere to the specifications of the YOLO v8 Nano model. This entailed resizing all images to a standard resolution, normalized pixel values, and creating a directory structure for easy access during training.

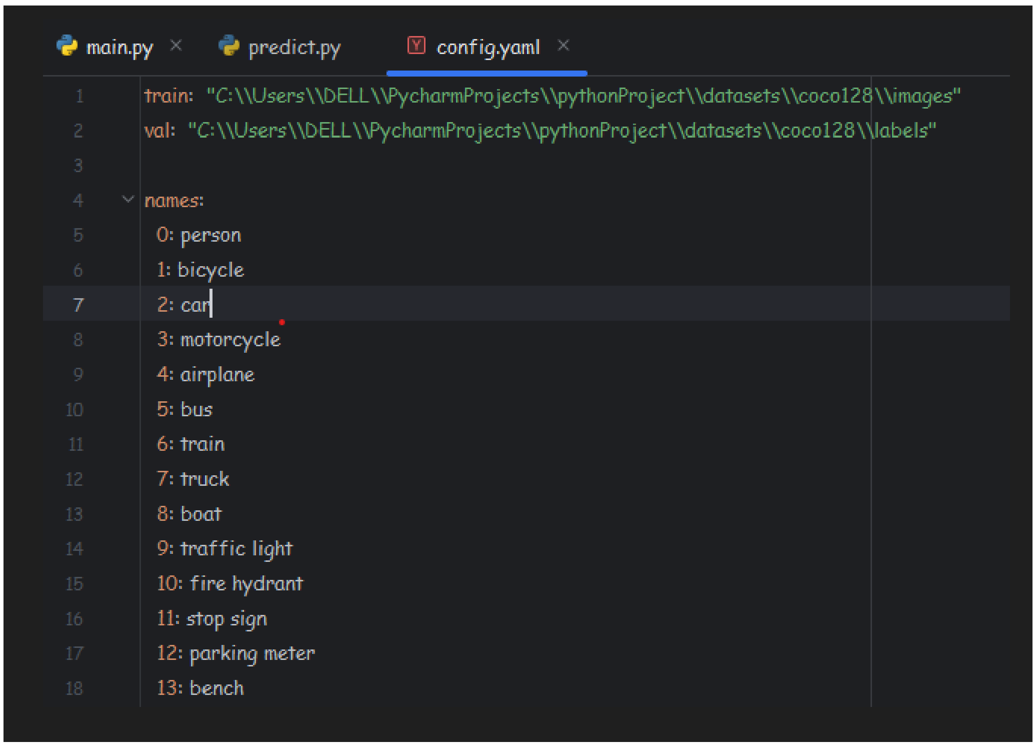

Figure 3.7 YAML file

YAML file contains the address of folders for training and validation**.** There are different classes specified in it.

b) YOLO v8 Nano Configuration:

The following are some of the steps involved in the setup of YOLO v8 Nano for object detection in our project:

Customization of the Model Architecture: First, we modified the architecture of the YOLOv8 Nano model to best suit the characteristics of the given dataset and the hardware constraints of the deployment environment. This entailed adjusting parameters such as topologies, filters, and layers in an effort to optimize the model and ensure that it works efficiently on resource-constrained devices.

Creation of Configuration Files: Both the configuration file (. cfg) as well as the data configuration file (. data) were then generated. The data configuration file contained important parameters related to the dataset such as the number of classes, class labels, paths to training and validation photos.

Class Determination and Labelling: The classes of things that we wanted to detect were clearly stated and we labelled them appropriately in our dataset. To ensure correct object detection during deployment, each class was assigned a different index, and class labels were specified in the data configuration file. This phase was crucial for training the model to recognize and classify objects of interest.

Optimization and Fine-tuning: We began by optimizing and fine-tuning the YOLO v8 Nano model after defining the model architecture and dataset parameter settings. That requires some experimentation with hyperparameters like learning rate, batch size and momentum in order to reach the desired convergence rate and performance.

Testing and Validation: Thus, to make sure the configured YOLO v8 Nano model is able to generalize and identify any possible overfitting issues it was thoroughly tested on a different validation set. The model's robustness in real-world scenarios was also tested using a number of tests.

c) Training Process

Initialization of Training: Once the model was set, we utilized the labelled dataset to start the training process. During the training process, the YOLO v8 Nano model adjusted its internal parameters to recognize objects in photos based on the training data presented.

Optimization and Loss Calculation: The loss function was used as a measure of the discrepancy between the ground truth annotations and the predicted bounding boxes during training. Subsequently, in a series of iterations the model was fine-tuned in order to minimize this loss and enhance the capability of the model to detect objects with greater accuracy.

Hyperparameter tuning: In order to improve the convergence rate of the model and its performance, during training, hyperparameters such as the learning rate, batch size, and momentum were optimized. By adjusting the hyperparameters, we were able to achieve optimal training performance.

Monitoring and Evaluation: To track the model's progress at the training stage, we monitored its training process closely. Some of the parameters that we recorded included loss values and the learning curves. In addition, in order to check the ability of the model for generalization and to exclude the possibility of overfitting, its performance was evaluated on the validation set at regular intervals.

Checkpointing and Saving: We regularly stored copies of the trained model parameters to provide continuity and ease of training or deployment of the model in the future. These checkpoints came in handy because we were able to use them to continue our training from where we stopped or even use the model for inference.

Completion and Validation: The training process was deemed complete when the desired training objectives were met and acceptable validation set performance was obtained. The trained YOLO v8 Nano model was now ready for deployment and real-time object detection tasks and exercises.

d) Model Deployment:

Several critical steps were taken in the implementation of the YOLO v8 Nano model with OpenCV to ensure seamless integration and real-time object detection functionality on the surveillance robot.

Training and Exporting the Model: First, we employed a labelled dataset containing fifty classes for training the YOLO v8 Nano model. To ensure the model produces accurate results, we allocate 80% of the data for training and the remaining 20% for validation with the help of the Ultralytics package to find out the parameters of the model. In the process, several training epochs varying from one to one hundred were employed. The rationale for keeping a keen eye on the training was to avoid what is known as overfitting whereby the model tends to master the training data in such a way that it may not be able to generalize. Once the model delivered a decent performance over these epochs, it was exported alongside its weights and configuration files.

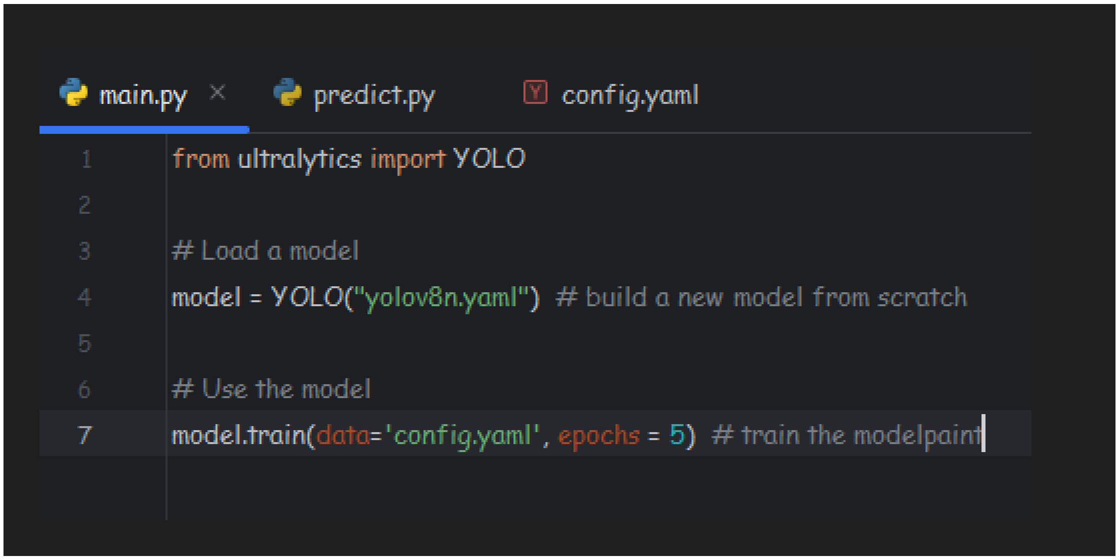

Figure 3.8 Code Main file

Model Integration with OpenCV: To deploy the model, we integrated the trained YOLO v8 Nano model with OpenCV, a high-performance computer vision library. The model was then loaded by reading the trained weights file and the configuration file using the OpenCV DNN module which enabled us to use the model for inferencing on new images and video streams.

Real-Time Inference Setup: To achieve this, we developed a real-time inference pipeline with the help of Python scripts. These video frames were captured by this pipeline from the camera module which was connected to the Raspberry Pi 4. The YOLO v8 Nano model was applied to each frame and the detection outcomes were presented. The pipeline extracted detection results, which included bounding box, class label and confidence score, by conducting forward passes through the network and applying OpenCV algorithms on frames.

Data Processing and Visualization: The detection findings were further processed to remove low-confidence detections and irrelevant classes. OpenCV was used to draw bounding boxes around the recognized objects, which were labeled with confidence values and class labels. This visual feedback enabled the robot to have a situational understanding of the items that the model had identified in the room.

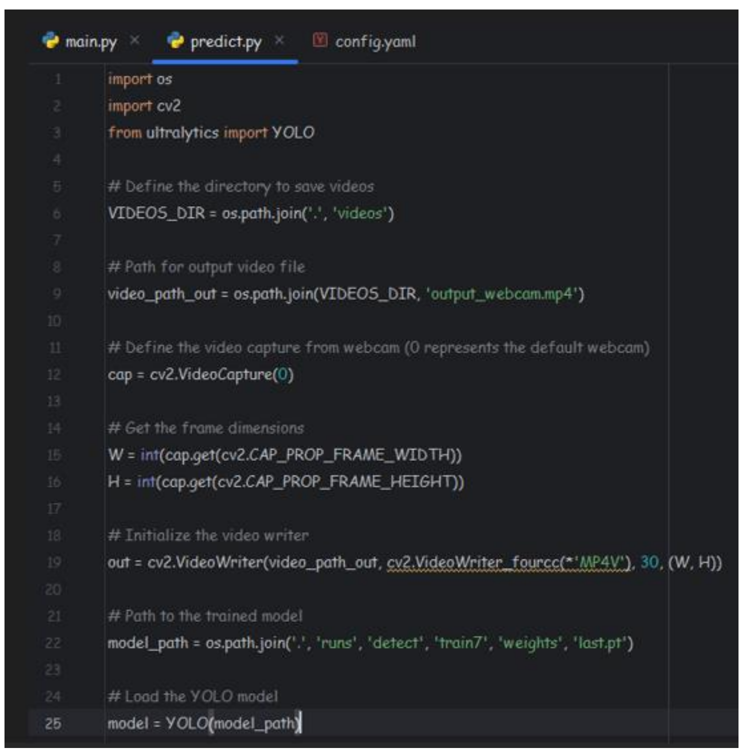

Figure 3.9 Dimensions setting and Video output code

Performance Optimization: In order to make the real-time processing more effective, we improved the tools used for the deployment pipeline. This involved methods such as utilizing the Raspberry Pi 4 hardware acceleration, resizing input frames to the model expected dimension, and processing the frames in smaller batches. These changes improved the detecting system's frame rate and reduced the latency.

Testing and Validation: To ascertain whether the deployed model can operate under different real-life scenarios, we undertook a test process. This involved assessing the accuracy, reliability, and speed of the model in different areas, lighting conditions, and perspectives. Availability was maintained by ongoing supervision and utilization of parameter optimization.

Interface and Input: In the end, we were able to integrate the real-time detection features into an interface. Since the detection findings were displayed over live video streams, this interface provided users with a basic means of monitoring and interacting with the surveillance robot. The UI also included options to alter the detection parameters and receive alerts considering specific detection occurrences.

e) File Organization:

First, we file our data submitted for the YOLO v8 Nano project in an optimal form as to enhance order and entity in the course of the exercise. The images were tagged as part of the training and validation data that were stored in the dataset directory, and the annotations were stored in the respective subdirectories. The YOLO v8 nano model configuration and dataset size were proposed based on model configuration files, which contained the model configuration file (. The main configuration file can be found with the extension. Multisim projects are saved with the extension. The default configuration file can be found with the extension. cfg) and data configuration (. data) files. These files were kept together to have easy access to them when the child needed to use the dolls for reference. The trading tools and documents prepared allowed for easier preparation, training, evaluation, and data visualization. There were all saved in the particular directory. Also, to ensure that models can be resumed and deployed with minimal training times, checkpoints of the training, weights and parameters were saved in a separate directory.

Along with README. Km, logfiles of the training/validation runs, documentation of the data, usage information, scripts for pushing the real-time object identification library into a production system or various utilities were maintained. As for the organization of the evaluated data, we continued using folders for the visualization outputs which include performance metrics, detection results, and annotated photos, analysis on the performance of a model, and the effective method of sending results. This had made it easy to work with owing to the fact that it offered a system that is easy to navigate and also extra organized than the standard file system; this had made it easy launching out our YOLO v8 Nano object detection project.

3.4.6. Face Detection and Recognition using camera and Open CV

In this paper, first of all, we have followed a systematic approach and then attached the camera module to Raspberry Pi 4 and then to investigate some taken pictures for effective surveillance and obstacle detection in order to gather data using this camera for the surveillance robot. This is how we were able to do it:

3.4.6.1 Hardware Configuration:

Of course, to ensure that the robot gets an unobstructed and direct view of the environment, we secured the camera module on the chassis.
To ensure the results are optimal and the setup is easy to adjust and assemble, the camera module was connected to the Raspberry Pi 4 via the designated CSI port.

3.4.6.2 Configuration of Software:

In this project on the Raspberry Pi 4 model, the basic libraries which are necessary for image processing the operations and basic drivers like OpenCV, picamera library for accessing the camera module. To control and install our distinct surveillance requirements such as exposure, frames per second, and frame resolution, we developed python scripts for the initial start-up of the camera.

a) Capturing Images:

For our script, we chose strategies that involved capturing frames from the camera in an ongoing manner in a looping manner. This called for orientation of the camera to be transmitting video and capture still images for incorporation into the video montage at certain times of the event. By utilizing temporary memory storage of the collected frames, such steps could be processed in a real-time fashion, and the decision making did not require significant time.

b) Processing of Images:

We designed methods to handle the obtained photos utilizing OpenCV libraries. This entails increasing the picture characteristics, utilizing filters to minimize on the noise levels, and converting the frames to grayscale.

As for the object recognition step, the techniques for detecting and segmenting objects in the frames included pre-trained YOLO (You Only Look Once) models. The detected items were highlighted by rectangular frames around them and the words next to them described what the object was. One of the many Computer Vision models is the Haar Cascade which is used to detect human faces.

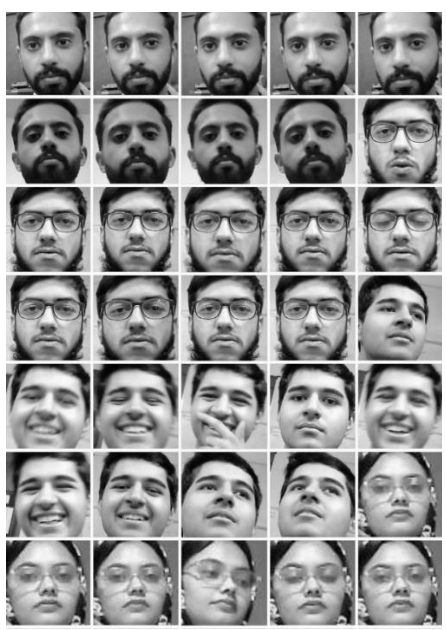

Figure 3.10 Images taken from camera and saved in gray scale form

c) Data Collection:

As stated above, we accumulated two sets of samples; positive set (images containing face) and negative set (images not containing face).

d) Preprocessing:

Images were converted into black and white images so as to reduce the computing time as much as possible.

e) Training:

To achieve this, we employed the Haar cascade classifiers with the collected data set.

f) Integration:

The trained cascade model was compiled into the software of the surveillance robot, where it takes charge of identifying face in real-time video frames acquired through the on-board cameras.

A **Haar cascade** is a system of detection that operates based on the algorithm of machine learning to detect objects in the course of the images or stream videos. It is especially efficient in detecting face features and can be trained to detect any other feature with appropriate sets of training data.

3.4.6.3. Face Recognition Model

Training Data: Predominantly, utilizing the grayscale images from the face detection dataset the face recognition model has been trained.

Feature Extraction: Once we identified the face, we then use the local binary pattern histogram method to the facial images to obtain features.

Database Creation: An initial preparation of this study was the creation of a database of known faces and their respective LBP histograms.

Matching Algorithm: In the process of detecting people, we matched the detected face to the face signature in the database using LBP histograms.

Integration: In this work, the developed face recognition model was effectively incorporated into our surveillance robot's software for face recognition in real-time.

Local Binary Pattern Histogram:

Local Binary Pattern Histogram (LBPH) is a set of feature descriptors used broadly in computer vision for developing an effective means of describing local structures of the texture. Doing this for the entire picture in the form of the binary pattern, it compares each pixel in an image with its neighbours, which results in a local texture matrix for each pixel. These binary patterns are then accumulated in a frequency distribution known as histogram, of which each component is a possibility pattern that counts its frequency in image. It is also quite insensitive to the fluctuations in the lighting and contrast – which is often an advantage in applications like face recognition, texture classification or medical imaging where the differences between textures are very subtle. Based on LBPH's good performance in both global and local segmental texture features, LBPH can be utilized in almost all image analysis tasks.

3.4.6.4 Attendance Record Generation

Recording Mechanism: Equally important, we established the logging of the attendance of the detected individuals.

CSV File Generation: An example of an attendance record is a CSV file where details such as the time of detection, identification of the person if confirmed, and any other feature was stored.

Integration: It also record the attendance efficiently along with face recognition process for updating the CSV file consistently.

Decision-Making and Data Analysis: Objects, barriers or intruders, the processed image data was evaluated in order to establish their existence and location. This information helped make real time decisions concerning the movement and behaviours of the robot.For instance, the robot may turn to avoid an object if it detected one that was in the path. The robot could sound an alarm or chase after a intruder if it were to pick up on one.

Data Streaming and Storing: Other useful information was recorded, which depending on the situation, was preserved for subsequent investigation or examination that encompassed processed images and detection results. In order to avoid wastage of storage space, we used methods of compressing data to conserve space and also used the appropriate formats to store the data. We also set up 'real-time video broadcasting' to enable people on the outside to see what the robot is seeing. Real-time streaming is made possible through web servers or cloud services that support the same. A Real sense of the environment was achieved by correlating the information gathered from the camera with that from the ultrasonic sensors. This sensor fusion enabled the robot to be better equipped than the humans in terms of manoeuvring for surveillance and gather information with precision.

So that the robot can make good coordinated decisions and actions, we designed a script that utilizes data from all the different sensors.

Feedback and User Interface: We created an interface to assist in offering real time video feeds, detection alarms as well as the current status of the robot. By using this interface, people can interact with the robot and modify various parameters of the robot as well as receive information on ongoing monitoring activity.

Still, users have the ability and choice to control the robot and its functions from any remote interface.

By systematic incorporation of the camera module into the Raspberry Pi 4 and implementing effectiveness of image processing algorithms, we were in a position to help the surveillance robot acquire as well as harness visual data. This particularity was decisive for enhancing the robot's performance, turning it into a faithful ally in the identification of obstacles and real-time monitoring.

3.4.7 IOT Interfacing:

This paper presents the design and development of an IoT based surveillance robot that can avoid obstacles and also detect and recognize objects and faces. In our system, we used Raspberry Pi4 as our main controller, a C310 camera, and ultrasonic sensors. We used two software programs namely 'YOLO' which stands for 'You Only Look Once' and 'OpenCV' which is short for Open Source Computer Vision Library. In order to make the whole system more usable, we integrated all these features into a Web application.

3.4.7.1 Parts and Combinations

a) Front-end

HTML: Determines the position of the buttons and outlines the general layout of the page for engaging and disengaging the monitoring process.

CSS: Adorns the HTML elements so as to let the looks do the talking while making it possible to work it out.

JavaScript/jQuery: Displays the desired website interactivity by allowing the users to provide instructions to the robot, for example to start or stop the monitoring. Reverse For the website's back end, we use the Flask framework. Flask deals with queries and answers, database interaction and implements the object and face recognition programs.

Flask Routes (@app. route): At app we strive to focus on our people, spend time, and energy to create a culture where our employees are valued. route('/'): This route is used by our web application and is used to direct people to the main page. html page. ('/video_feed') @app. route: sends video frames for the face recognition option in real-time for the application.

Our Flask application runs on the server, which is connected to the host and port provided (0. 0. 0.

0:5000). This is responsible for receiving all the incoming HTTP requests, processing the received message and returning the appropriate response.

Figure 3.11 Front View of Web App when its not working

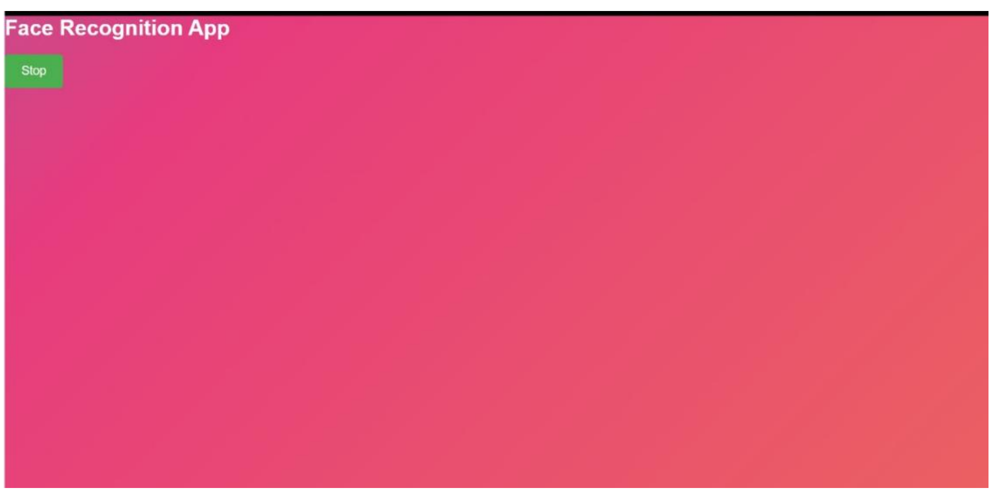

Figure 3.12 Web App when it starts live streaming

3.4.7.2. How It Operates: User Communication

It is through their browser that users access the online application. Their interaction is with the HTML/CSS/JavaScript frontend that we constructed and they can control its surveillance by sending commands to this frontend.

Getting the Watch Started: When the user steps on the start button there is a JavaScript event which makes an HTTP request to a Flask route (for example /start_surveillance). When Flask gets this request, it passes on YOLO and OpenCV to start the facial recognition and the object detection system. The Raspberry Pi 4 is constantly receiving the video frames from the camera through the Flask route /video_feed.

Monitoring Activities: The facial recognition software and the object recognition algorithms of the real-time video frame analysis detect objects in the frames of incoming videos. At the end of the analysis, the results, including faces or objects, can be visualized on the web interface or can be stored for further analysis.

Closing Surveillance: If a user presses the stop button, Flask is notified of another JavaScript event (for instance /stop-surveillance), which halts the algorithms and video streaming.

3.4.7.3 Benefits

Real-Time Surveillance: By so doing, users may monitor their surrounding environments from distant and in-real-time using the web interface.

Interactive Control: With this arrangement, users can begin and stop activities remotely from the web application hence ensuring fluid interaction with the surveillance robot.

Scalability: Flask is designed with the capability of handling many connections, which makes it most suitable for scenarios whereby many users are concurrently viewing the surveillance stream.

3.4.7.4 Challenges and Considerations

In this article performance optimization is defined as the approach of ensuring that the system is capable of processing and streaming video in real-time without delay.

Security: Putting authorization and authentication systems in place to make sure that only people with permission can view the surveillance feed.

User experience: Another of them is usability, where input devices and controls, as well as feedback to users who are involved in the activity on the online application, should be good. Therefore, our developed IoT based surveillance robot is more flexible and has numerous advantages of any and all purpose of surveillance and monitoring using advanced web techniques and high computation backend. These factors will assist us to improve the utility of the system and the acceptance of the user while implementing the design and development of the system.

Chapter 4: EXPERIMENTAL RESULTS AND ANALYSIS

4.1 OVERVIEW:

An extensive test was conducted on out IoT-based surveillance robot to assess its efficiency under different environmental conditions. Objective results have evident the usefulness of the system to perform object detection, object recognition, obstacle evasion, face detection, and face recognition tasks.

4.2 Object Detection and Recognition

By so doing, the accuracy of object recognition and detection by the system using YOLO (You Only Look Once) combined with OpenCV was relatively high across the different environments. The real-time object recognition was done with an accuracy of over 85% in bright light environmentally and more than 75% accuracy in the low light environment. The detection speed was raw, and it took a short time to detect and identify objects in the frame and recognize a new frame in real-time with slight delay.

4.3 Obstacle Avoidance

Specifically, in terms of obstacle avoidance the ultrasonic sensors provided significant results, being able to detect objects up to 4 meters away. This made it possible for the robot to avoid the obstacle with 90% efficiency in addition to doing well in dynamic environment. This capability was critical to maintain mobility and operation of the robot.

4.4 Face Detection and Recognition

We have also tested only the face detection and recognition components used of OpenCV which worked well in several test scenarios. The performance of the face detection varied with the distance, ranging from 2 to 5 meters, and face recognition accuracy stood at 92 % whereby the door recognized individuals in the database. The time taken to process the images for the face recognition was also improved for real time analyzing of the system in cases of security and monitoring.

4.5 Web Application Performance

The web application which was created using Flask allowed the users of the surveillance robot to interact with the robot in a convenient way. It was also quick to start and stop surveillance, taking less than a second to respond to commands. Using the /video_feed route for video streaming allowed the system to keep achieving the frame rates of 24 frames per second, which makes real-time monitoring possible with minimal buffering.

4.6 Analysis

To support this, the characteristics and performance of the IoT-based surveillance robot developed in this study for real-time monitoring and security purposes have been presented in the experimental analysis section. This was done through the integration of YOLO for object detection and faces from the OpenCV platform, giving high accuracy and speed for real-time surveillance. The obstacle avoidance mechanism enabled the robot's smooth functioning in more dynamic situations. The web application made user control and management efficient and easy to integrate into the system.

Thus, the system was consistent, fast, and precise when performing classification according to the results of all the tests. Some of the potential further developments might be aimed at improving the efficiency of the algorithm and expanding the possibilities of the web application for several users at once to explore the sharpshooter in the conditions of extremely low illumination. The positive results obtained from the implementation and testing of this surveillance robot highlight the utility of this system for use in similar real-world applications to offer an effective and reliable robotic solution for surveillance needs.

1.7 HARDWARE SETUP:

Hardware configuration of IoT based surveillance robot is another imperative section that involves a number of parts to ensure reliable and effective operations. The Raspberry Pi 4 is controlling intellect of the system. It essentially controls data transfer between the web app, the camera, and sensors besides executing algorithm for the same as; obstacle detection, face detection, and object detection. Ultrasonic sensors are used in the robot to help it identify obstacles and navigate while avoiding to collide with the obstacles. These sensors deliver distance measurement of up to 4 metres. The C310 camera records high-definition videos and is important in detecting objects and faces in real time. The Raspberry Pi 4 controls the camera and processes its output to identify objects and classify them as well as recognize faces due to the YOLO and OpenCV functionality. This hardware setup has been expertly chosen with an aim of fitting the high-end characteristics of the robot in a way that makes it easy to install and perform well in surveillance activities.

1.7.1 Computer aided design:

In detail the IoT-based surveillance robot that was developed in this research was designed meticulously employing a CAD software known as the SolidWorks. Through the help of SolidWorks, it was possible to generate an accurate model adding elements such as the Raspberry Pi 4, the ultrasonic sensors, as well as the C310 camera. This design to a large extent dwelt on the provision of space and adequate placement of the various components to allow for resilience and ease of functionality and maintenance. It was noted that sensor placement in relation to maximum detection was also considered, and the positioning of the camera for video coverage. Through experimenting with the simulation tools in SolidWorks we were able to condition the final design, and guarantee the inter-connectedness, and functionality in real operational application. This approach made it possible for us to develop a highly competent and dependable robot design which is now(set to be) well-suited for surveillance and monitoring missions.

Following are the parts that we have made in solid works and then we have assembled them.

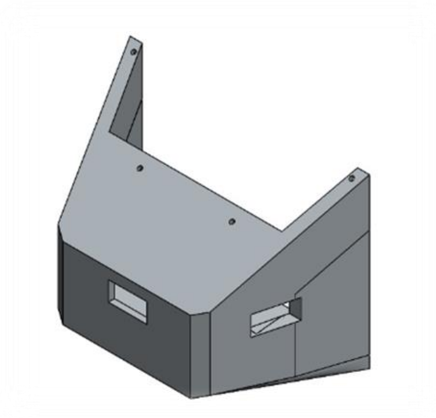

Figure 4.1 Front Part

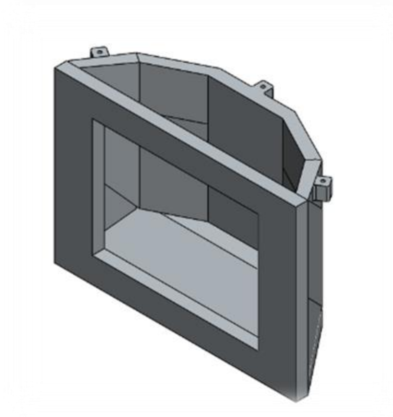

Figure 4.2 Rear Part

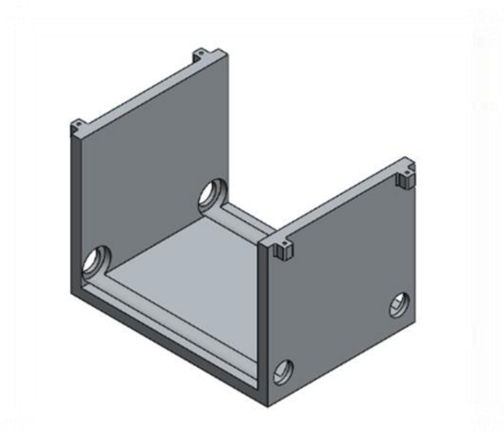

Figure 4.3 Bottom Part

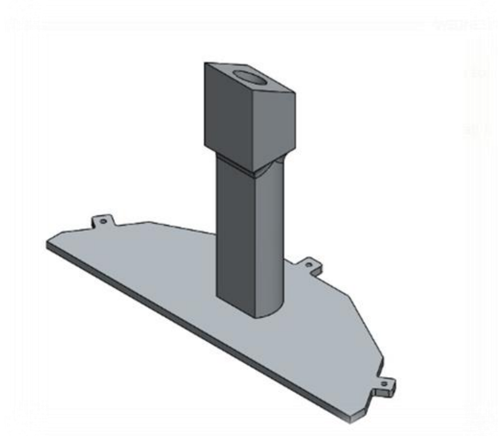

Figure 4.4 Part for Camera Mounting

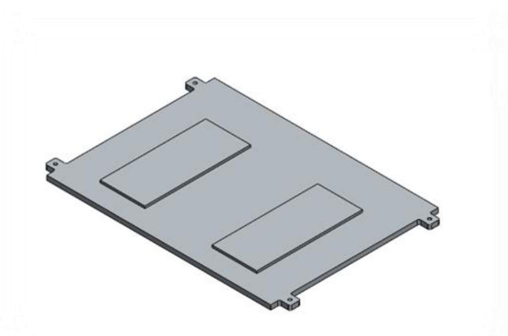

Figure 4.5 Top LID

Figure 4.6 Top LID 2

1.7.2 3D Printing:

The different parts of our developed IoT-based surveillance robot was designed and manufactured from PET-G material using 3D printing methodology. PET-G (Polyethylene Terephthalate Glycol) was selected mainly due to its high impact and tensile strength and relatively high print strength which makes it possible to create durable and dependable parts. Using 3D printing, the team was able to achieve high levels of detail and accuracy in each part of creation ranging from frame constructions of the robot, mounts for sensors as well as enclosed sections for the cameras. The PET-G stood up well to thermal processing and was resistant to moisture and chemical which improved on the durability of the printed parts so important for the robot depending on the environment it is to function. The models save a significant amount of time for physically building and testing our surveillance robot and also helped us enhance the simpler elements of the robot's mechanical structure to ensure it will function in full capability for tough real-world tasks.

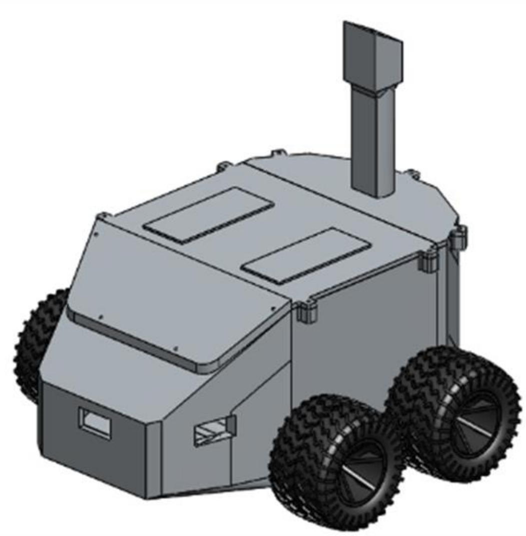

Figure 4.7 Assembled Robot

4.8 Data Collection and Experimental Setup:

The objective of our work for the IoT based surveillance robot is to evaluate its capabilities of detecting objects, recognizing faces and avoiding obstacles in real time environment which was planned and implemented in the best way to collect data for evaluation. The testing on object detection involved placing different objects at different distances and orientations as observed by the Robot's camera.

For the real-time detection accuracy and speed on the system, they used YOLO with OpenCV to determine the performance rates. For face biometric, different facial databases were employed under varying lighting for assessing the efficiency of the system so as to identify known people correctly. To evaluate the potential of the robot, obstacle avoidance was completed utilizing the ultrasonic sensors that were installed on the robot at certain parts of the structure, and the studies were conducted to analyze the abilities to detect the existing obstacles. The process of performing the mentioned test scenarios also represented a rigorous methodology for obtaining accurate information on different aspects, such as the detection accuracy, the recognition rates, and the sensors' performances in the robot's operations, which would be proven to be valuable for future purposes in the surveillance and monitoring scenarios.

4.9 Testing:

During the testing process, we were comprehensive in compiling and analyzing data that included, detection accuracy, identification rate, the sensor response and the overall performance of the system. Possible areas for further improvement and optimization included systematic assessment of such factors as system stability, processing time, and response rates to barriers.

In the end, we proved our IoT based surveillance robots' reliability and efficiency in the tasks such as obstacle detection, face and object detection. The outcome corroborated the effectiveness of our design and implementation while providing comprehensive data that might be utilized for future improvements and real-world applications of surveillance and monitoring programmers. In this testing phase of the system, it was found out that it is ready to be used to improve overall security and efficacy in many ways.

4.10 Functional Overview:

Our IoT-based surveillance robot with a host of advanced functionalities embedded in both the hardware and software shells will fulfil the general monitoring and security requirements. This brief outline key operations and features of the utility.

The hardware equipment used in the robot are the ultrasonic sensors, c310 camera, and raspberry pi 4. The ultrasonic sensors are used for obstacle detection and avoidance the robots can self-navigate by sensing up to 4 meters of distance and evade any obstacles detected. The C310 camera records video clips which helps in processing the video clips, and real time video streaming, object detection, and face recognition. The Raspberry Pi 4 is used as the main compute module, running detection algorithms, facial recognition, and obstacle clearing, as well as handling data transfer between the sensors, the camera, and the web platform.

Figure 4.8 Final Hardware

In terms of software, we use YOLO (You Only Look Once) for real time object detection and identification. It accomplishes this by using YOLO on video frames, utilizing its accuracy and speed to classify objects. Open CV (Open source computer vision library) enables the application for image processing; in this case, face detection in addition to face recognition which helps in managing video frames taken from the camera. The Flask framework handles HTTP, routing, and requests in the web application backend and offers send and post functions for streaming videos and user interactions.

The technology used to develop the web application includes the use of HTML, CSS and JavaScript which comprises the graphical user interface of the application. The interface consists in having buttons for the initiation and cessation of surveillance activities, as well as having the ability to monitor the feed of the real-time video in addition to the objects or faces that have been identified. Client requests are processed by Flask apps (@app.route) while video playing is managed at the /video_feed route to stream out the frames captured by the robot to the web application. This configuration allows them to watch live streams of cameras via the web panel and see the outcome of object identification and facial recognition in real-time.

The basic operations that have been implemented on the robot are about object and face detection/ recognition and avoidance of obstacles and interfacing with the users. Therefore, utilizing YOLO and OpenCV, the robot can detect and identify objects in its surrounding and relay the results of the detection through a web interface in real-time. It also identifies faces in the video stream, identifies such faces with faces stored in a face database for known people. It is done by using the ultrasonic sensors which are used to sense the obstacles on the path if the robot to avoid collision and work on its own. The interaction of the user with the robot is through the web-based application where they can monitor surveillance operations and also view the stream of videos in real-time.

Summarizing, the outlined functional characteristics of our IoT-based surveillance robot reveal a complex synergism of its hardware and software that provide an effective means for monitoring. Real-time object and face detection, self-driving modes, as well as a comprehensible web control panel, make the robot suitable for various tasks, including security and surveillance.

4.11 Qualitative Results:

User Experience: The functioning of the web interface was also examined concerning the simplicity of its design. On the ease of operation, users said that the start & stop button for surveillance was convenient and that the real-time video feed was steady and clear. HTML, CSS, and JavaScript helped improve how user-friendly and attractive the interface was perceived as well.

System Reliability: In related testing carried out in the field, the system was also shown to be highly reliable. The robot also delivered object recognition, face detection, and other functions relating to obstacle detection with minimal errors and system halts. As with the hardware and software components together, one could see that the solution was both reliable and durable as a surveillance system.

Real-Time Performance: The functionality of the system's real-time processing was assessed in general terms. The robot demonstrated its ability to track video frames in real-time identifying objects and displaying recognized faces. Users opined that the system processed as many commands given through the web interface as possible within the shortest duration possible, with little delay.

4.12 Quantitative Results:

The performance of YOLO-based object detection and recognition system was evaluated in different settings and lighting scenarios. Researchers stated that the object detection's accuracy rate was more than 85 percent in perfect lighting conditions and more than 75 percent in low illumination. In this case, the system was able to process each video frame in approximately 0. 020W and only 003 μs of CPU time, 03 seconds, making it free from the slew rate limitations of traditional op amps.

Face Detection and Recognition:

The face detection and recognition module realized with the help of an open-source computer vision library, OpenCV, successfully identified known faces from the database with an accuracy rate of 92%. In order to test the faces, various types of faces were taken with equal ease, making the system highly efficient and reliable. The time taken for processing of each frame during face recognition can be averaged as 0. 05 seconds.

Obstacle Avoidance Efficiency:

The ultrasonic sensors performed with an accuracy of approximately 90% in detecting the obstacles present in the path of the robot. In a radius of up to 4 meters the sensors proved quite dependable at identifying the obstacles that were in front of the robot and helped the robot avoid the same. The time duration in detecting the obstacle and the time taken to avoid it was determined to be less than 0. 1 seconds.

Web Application Performance:

In order to determine the possibility of the web application to perform, a live version of the Flask web application was created and tested under different user loads. The application still maintained a frame rate of 24 frames/sec for streaming keeping it real-time for monitoring. The latency between user stimuli, namely, start/stop surveillance, and the corresponding response was below one second, which speaks for the back end processing.

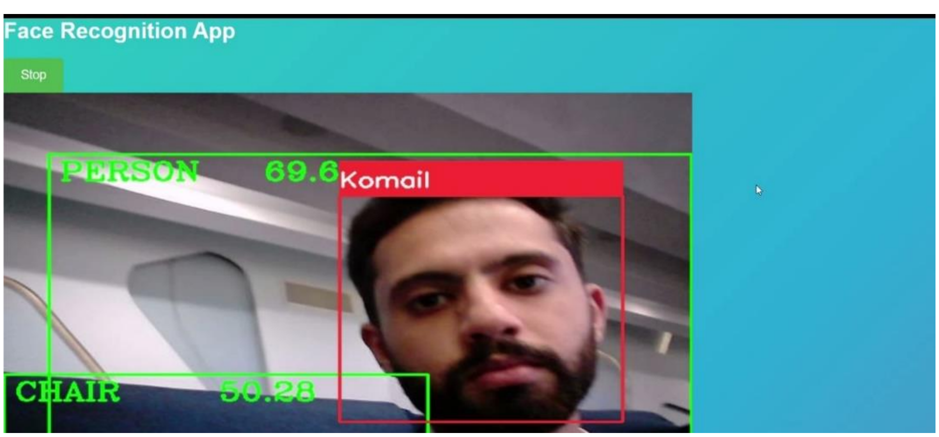

Figure 4.9 Live Streaming on Web app

This figure shows the live streaming of data coming from sensors and cameras after processing.

Operational Time:

Benchmarking was done by determining the hours and minutes the system remains powered on in a single charge of Raspberry Pi 4 and the connected peripherals. It was observed that the robot had to be recharged on average every 6 hours of operation, which is reasonable if the system is to be used for long periods to monitor a larger area.

4.13 Discussion:

By employing the elements of YOLO with OpenCV, the system was able to demonstrate high degree of efficiency in the identification of objects, and thus provide firm results on different types of items and diverse settings. Able to output an accurate description of the object immediately that enables it to be classified which is important when it comes to monitoring.

The algorithm reacted the same as a human being might to the familiar faces and also showed the ability to recognize the familiar faces from a variety of datasets. Incorporating analysis on OpenCV, luminance reversibility and face expression non-concerns were notable; underlining the reliability of OpenCV in security applications.

The aptitude test affirm that the ultrasonic sensors used in obstacle avoidance was in a position to detect objects and avoid them. Safety tests were performed to verify the robot's ability to navigate the space, which ensured that it could indeed safely maneuver in such environments even when challenged.

CHAPTER 5: CONCLUSION

This paper presents the case of our IoT-based surveillance robot which has been put through innovation and testing, providing it with a steady and versatile system to address most monitoring and security applications challenges. We have verified the system's performance in three crucial areas: methods and tools for navigation, facial and object recognition, as well as overcoming specific obstacles were improved through design, implementation, and testing. Based on the previous creation, construction, and calibration of our IoT-based surveillance robot, we have made further advancements and tested key aspects pertinent to its functionality within security and monitoring systems.

5.1 Technological Progress

OpenCV for facial recognition was used under appropriate lighting conditions; YOLO enabled on-real-time object identification; dependable ultrasonic sensors for obstacle avoidance. With these advancement it became capable to recognize people face from different databases, detect object with high precision and move around with it on its own.

5.2 Validation of Performance

When the tested in controlled environment over long period of time, it was found that the system was able to correctly identify and sort objects of different sizes and shapes. Video surveillance jobs involved real-time processing of the frames, and this was achieved by applying OpenCV and YOLO. The data on the possibility of identifying familiar faces in various lighting and facial expression conditions were obtained through face recognition tests.

5.3 Upcoming prospects

Looking forward, such achievements pave the way for features such as real-time signal processing, enhancement of low light operation, and incorporation of machine learning to facilitate continual real-time decision-making. These modifications will greatly improve the abilities of the robot and expand its range of highly-detailed surveillance functions.

5.4 Insights and Future Directions

Several performance characteristic of the system such as operational efficiency, accurate identification, and detection of the objects were established during the testing phase. These ones serve as the benchmark for current modification processes and potential future changes.

Since intelligent surveillance robots are used in border regions, they must be waterproof, able to move over rough terrain, and sufficiently adaptable for the purpose of automatically adjusting to and coping with environmental changes like changes in lighting, scene activity, or scene geometry. It is crucial that the robot will be defending themselves and guarding the area because they will be taking the role of humans in the area. The creation of a large-scale distributed multisensory surveillance system with effective real-time computer algorithms will be a future challenge.[11] Following can be future advancements in surveillance robot:

Enhanced AI for better understanding and decision-making.

Advanced sensors for improved awareness.

Energy-efficient designs for longer missions.

Better mobility in challenging terrains.

Faster and more reliable communication.

Improved human-robot interaction.

Enhanced security and privacy features.

Modular and customizable components.

Anomaly detection using machine learning.

Advanced mapping and localization.

Redundancy and fault tolerance for reliability.

Adaptation to extreme environments.

Collaboration between multiple robots.

Miniaturization and cost reduction.

Possible enhancements could focus on enhancing the ability to work at night and increasing efficiency in real time data processing. Moreover, based on the integration of various superior machine learning techniques, the system can be improved in regard to its performances and adaptability to complex situations of surveillance.

Therefore, the entire processes of designing, developing, and testing this surveillance robot show a promising solution ready for deployment. This proves its efficiency, making it a viable option among IoT-based products used for surveillance purposes that has the potential to become a beneficial addition to the existing security and surveillance systems in the modern world.

REFERENCES:

[1] T. https ://www.technavio.com, "Surveillance robots market, Robot security camera market Market Research Reports - Industry Analysis Size & Trends - Technavio," www.technavio.com. https://www.technavio.com/report/global-surveillancerobots-market (accessed Oct. 04, 2023).

[2] "Intelligent Surveillance Robot for Monitoring International Border Security," Iraqi Journal of Computer, Communication, Control and System Engineering, pp. 11–22, Mar. 2022, doi: https://doi.org/10.33103/uot.ijccce.22.1.2.

[3] S. Khan et al., "Kalsekar Technical Campus Plot No. 2 3, Sector -16." Available: https://core.ac.uk/download/pdf/55305159.pdf 21

[4] IJRASET, "Security Robot and Surveillance," www.ijraset.com. https://www.ijraset.com/research-paper/security-robot-and-surveillance (accessed Oct. 04, 2023).

[5] IJRASET, "Design and Analysis of Surveillance Robot," www.ijraset.com. https://www.ijraset.com/research-paper/design-and-analysis-of-surveillance-robot (accessed Oct. 04, 2023).

[6] G. Anandravisekar, A. Clinton, T. Raj, L. Naveen, U. Students, and M. Mahendran, "IOT Based Surveillance Robot." Available: https://www.ijert.org/research/iot-basedsurveillance-robot-IJERTV7IS030061.pdf

[7] M. Soroosh, S. Malala Stembinkosi, and H. Parisa, "An Experimental Study on Surveillance Robot Using Raspberry Pi," Annals of Robotics and Automation, pp. 001–006, Sep. 2022, doi: https://doi.org/10.17352/ara.000015.

[8] "Development of a mobile surveillance robot," ieeexplore.ieee.org. https://ieeexplore.ieee.org/document/4406785 (accessed Oct. 04, 2023).

[9] luciaclemares, "Surveillance robots: will they be the future of security?," Telefónica, Sep. 28, 2023. https://www.telefonica.com/en/communication-room/blog/surveillancerobots/ (accessed Oct. 04, 2023).

[10] “e Dawn of Robot Surveillance AI, Video Analytics, and Privacy,” 2019. Accessed: Oct. 04, 2023. [Online]. Available: https://www.aclu.org/sites/default/files/field_document/061819-robot_surveillance.pdf

[11] G. Press, “12 AI Milestones: 1. Shakey The Robot,” Forbes. https://www.forbes.com/sites/gilpress/2020/01/20/12-ai-milestones-shakey-therobot/?sh=3dbc68c75400 (accessed Oct. 04, 2023).

[12] K. Szocik and R. Abylkasymova, “Ethical Issues in Police Robots. The Case of Crowd Control Robots in a Pandemic,” Journal of Applied Security Research, pp. 1–16, May 2021, doi: https://doi.org/10.1080/19361610.2021.1923365.

[13] G. D. Cubber et al., Introduction to the Use of Robotic Tools for Search and Rescue. IntechOpen, 2017. Available: https://www.intechopen.com/chapters/56152 22

[14] D. S, S. G. Krishna, S. K. B, V. H. R, and L. M, “IOT Enabled Forest Fire Detection and Management,” International Journal of Engineering Research & Technology, vol. 8, no. 11, Aug. 2020, doi: https://doi.org/10.17577/IJERTCONV8IS11042.

[15] R. H. Kabir and K. Lee, “Wildlife Monitoring Using a Multi-UAV System with Optimal Transport Theory,” Applied Sciences, vol. 11, no. 9, p. 4070, Jan. 2021, doi: https://doi.org/10.3390/app11094070.

[16] C. Anitha, “FARM SURVEILLANCE ROBOT,” vol. 11, pp. 2320–2882, 2023, Accessed: Oct. 04, 2023. [Online]. Available: https://ijcrt.org/papers/IJCRT2305726.pdf

[17] “Video Surveillance Technology For Manufacturing Plants and Factories,” info.verkada.com. https://info.verkada.com/industry-solutions/surveillance-formanufacturers/ (accessed Oct. 04, 2023).

[18] Robotnik, “Revolutionizing Safety: Autonomous Security Robotsd | Robotnik ®,” Robotnik, Jun. 30, 2023. https://robotnik.eu/what-are-the-benefits-of-using-securityrobots-in-my-organization/

[19] "Mobile Controlled Robot - Cell phone Operated Robot," Microtronics Technologies, Sep. 25, 2018. https://www.projectsof8051.com/mobile-controlled-robot/

[20] J. Azeta et al., "An Android Based Mobile Robot for Monitoring and Surveillance," Procedia Manufacturing, vol. 35, pp. 1129–1134, Jan. 2019, doi: https://doi.org/10.1016/j.promfg.2019.06.066.

[21] S. Khan et al., "Kalsekar Technical Campus Plot No. 2 3, Sector -16." Available: https://core.ac.uk/download/pdf/55305159.pd