AI Intergration In Combat Vechicles

A PROJECT REPORT

Submitted by

PAKALA SIVASUBRAMANYAM SAKETH [REG NO: 211420243038]
SANTHOSH KUMAR S [REG NO: 211420243056]

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & DATA SCIENCE

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

MARCH 2024

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report **"AI Intergration In Combat Vechicles"** is the bonafide work of **" PAKALA SIVASUBRAMANYAM SAKETH (211420243038) & SANTHOSH KUMAR S (211420243058) "** who carried out the project work under my supervision.

SIGNATURE

Dr. S. MALATHI
HEAD OF THE DEPARTMENT,
Department of AI&DS,
Panimalar Engineering College,
Chennai - 123

SIGNATURE

Mrs. S. VIMALA
ASSOCIATE PROFESSOR,
Department of AI&DS,
Panimalar Engineering College,
Chennai - 123

Certified that the above-mentioned students were examined in the End Semester project (AD8811) held on __________

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION BY THE STUDENTS

We **Pakala Sivasubramanyam Saketh (211420243038), Santhosh Kumar S (211420243056)** hereby declare that this project report titled "**AI Intergration In Combat Vechicles**", under the guidance of Mrs.S.Vimala is the original work done by us and we have not plagiarized or submitted to any other degree in any university by us.

ACKNOWLEDGEMENT

I would like to express our deep gratitude to our respected Secretary and Correspondent **Dr. P. CHINNADURAI, M.A., Ph.D.,** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

I express our sincere thanks to our Directors **Tmt.C.VIJAYA RAJESWARI, Dr. C. SAKTHI KUMAR, M.E., Ph.D. and Dr. SARANYASREE SAKTHI KUMAR B.E., M.B.A., Ph.D.,** for providing us with the necessary facilities to undertake this project.

I also express our gratitude to our Principal **Dr. K. MANI, M.E., Ph.D.** who facilitated us in completing the project. I thank the Head of the Artificial Intelligence & Data Science Department, **Dr. S. MALATHI, M.E., Ph.D.,** for the support extended throughout the project.

I would like to thank our supervisor **Mrs. S. VIMALA**, coordinator Dr. **K.JAYASHREE & Dr. P.KAVITHA** and all the faculty members of the Department of AI&DS for their advice and encouragement for the successful completion of the project.

PAKALA SIVASUBRAMANYAM SAKETH

SANTHOSH KUMAR S

ABSTRACT

The proposed system utilizes Raspberry Pi 4B and ARDUINO UNO microcontrollers to improve existingIntegrating security and surveillance setups. a Pi Cam 8MP with Raspberry Pi 4B enhances intruder detection accuracy. Once detection occurs, a servo motor promptly adjusts a laser for precise targeting, boosting operational efficiency. Alongside this, an ultrasonic sensor, combined with image processing techniques, further enhances detection capabilities, accurately identifying threats. Additionally, GPS technology ensures reliable device localization, aiding operational effectiveness and facilitating system recovery in critical situations. The ARDUINO UNO microcontroller acts as the central command unit, managing system functions and essential programs, simplifying control and programming. These advancements provide the system with increased precision, versatility, and autonomy, surpassing the limitations of conventional security setups. By integrating state-of-the-art hardware components and intelligent algorithms, the proposed system has the potential to transform surveillance and security protocols, offering a robust defense against potential threats across various operational scenarios.

Keywords: Robotics, IOT. Raspberry pi 4b, laser, Ultrasonic sensor.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER - 1

INTRODUCTION

INTRODUCTION

The security robot project centers on the utilization of a Raspberry Pi 4B as its core component, specifically engineered to detect and identify intruders through the integration of a Pi Cam 8MP. This high-resolution camera serves as the primary data capture device, enabling the system to gather visual information crucial for intrusion detection. The captured data is then transmitted for processing, where sophisticated algorithms analyze and interpret it to identify potential threats within the robot's vicinity. To enhance accuracy in targeting potential intruders, a servo motor is employed to control a laser system, ensuring precise alignment and effective threat response.

In addition to its visual detection capabilities, the security robot incorporates a GPS module to facilitate efficient device localization and recovery mechanisms in the event of displacement. This feature not only enhances operational efficiency by providing real-time location tracking but also serves as a crucial asset for mission-critical tasks where the robot's precise positioning is paramount. Furthermore, the integration of an ultrasonic sensor further augments the robot's object detection capabilities, enabling it to navigate and interact with its environment with heightened precision and awareness. Combined with Python image processing techniques, these sensors enable the system to accurately discern potential threats and take appropriate action.

The security robot project centers on the utilization of a Raspberry Pi 4B as its core component, specifically engineered to detect and identify intruders through the integration of a Pi Cam 8MP. This high-resolution camera serves as the primary data capture device, enabling the system to gather visual information crucial for intrusion detection. The captured data is then transmitted for processing, where sophisticated algorithms analyze and interpret it to identify

potential threats within the robot's vicinity. To enhance accuracy in targeting potential intruders, a servo motor is employed to control a laser system, ensuring precise alignment and effective threat response.

In addition to its visual detection capabilities, the security robot incorporates a GPS module to facilitate efficient device localization and recovery mechanisms in the event of displacement. This feature not only enhances operational efficiency by providing real-time location tracking but also serves as a crucial asset for mission-critical tasks where the robot's precise positioning is paramount. Furthermore, the integration of an ultrasonic sensor further augments the robot's object detection capabilities, enabling it to navigate and interact with its environment with heightened precision and awareness. Combined with Python image processing techniques, these sensors enable the system to accurately discern potential threats and take appropriate action.

At the heart of the security robot lies an Arduino Mega microcontroller, serving as the central control unit responsible for coordinating the functionalities of its various components. Powered by a robust 12V battery, the system ensures uninterrupted operation, essential for maintaining continuous surveillance and security vigilance. The robot's mobility is facilitated by a sturdy chassis equipped with a motor driver and two DC motors, enabling it to navigate diverse terrains and environments effectively. Moreover, critical information and real-time updates are displayed on an LCD screen, providing operators with immediate insights into the robot's status and operational parameters.

Furthermore, the integration of IoT technology into the security robot enables seamless remote monitoring and control capabilities, enhancing situational awareness and adaptability to evolving mission requirements. This connectivity allows operators to remotely access the robot's functionalities, receive live updates, and adjust its operational parameters as needed. By leveraging cutting-edge hardware components and intelligent algorithms, the security robot project aims to revolutionize surveillance and security protocols, offering a formidable defense mechanism against potential threats across various operational scenarios.

1.1 PROBLEM DEFINITION

The objective of this project is to create an adaptable security robot proficient in identifying and neutralizing intruders. It employs a Raspberry Pi 4B processor alongside a Pi Cam 8MP, enabling the robot to detect potential threats and relay captured data for further analysis. Enhancing its defensive capabilities, a servo motor-operated laser system ensures precise targeting of identified threats. Integration of a GPS module not only supports efficient operation but also aids in recovery efforts if the robot gets displaced during its tasks.

Moreover, the robot's threat detection capabilities are bolstered by an ultrasonic sensor, while Python image processing algorithms refine the accuracy of threat identification. Acting as the central control unit, the Arduino Mega microcontroller seamlessly integrates and manages the various components of the system, ensuring optimal performance throughout operations. Real-time feedback is provided through an LCD display, allowing operators to stay informed of the robot's status and operational parameters.

Furthermore, the project incorporates IoT integration, enabling remote monitoring and control functionalities. This feature enhances overall security effectiveness by facilitating constant surveillance and the ability to adjust operational strategies as needed. By leveraging these advanced technologies and capabilities, the project aims to develop a highly versatile security solution capable of effectively addressing security threats across diverse operational scenarios.

CHAPTER - 2

LITERATURE SURVEY

TITLE 1: Modeling the Impact of High Energy Laser Weapon on the Mission Effectiveness of Unmanned Combat Aerial Vehicles

AUTHOR: QIJIA YUN , BIFENG SONG , AND YANG PEI

YEAR: 2020

DESCRIPTION:

Amidst the rapid advancement of high energy laser weapons (HELWs), there is burgeoning interest in integrating them onto unmanned combat aerial vehicles (UCAVs) to assess their impact on mission effectiveness. This paper proposes a comprehensive 4-level design framework based on system-of-system (SoS) oriented design to investigate this integration. The framework evaluates HELW impact across four key aspects: strike capability, stealth performance, vulnerability, and defense capability, while considering UCAV design constraints. Simulation experiments, conducted through agent-based modeling and simulation, focus on penetration scenarios. Results indicate that integrating HELW enhances UCAV survivability and mission effectiveness, particularly for high-speed, stealthy UCAVs. However, it's observed that the UCAV's mission effectiveness rate (MER) doesn't uniformly rise with increased HELW output power, underscoring the necessity to strike a balance between HELW and UCAV performance during conceptual design stages.

TITLE 2: A Collaborative Decision-Making Approach for Multi-Unmanned Combat Vehicles based on the Behavior Tree

AUTHOR: Shi Feng ,Junqiang Xi, XinXing Mu, Cheng Gong

YEAR: 2021

DESCRIPTION:

Addressing the challenge of effective behavior decision-making in multi-unmanned combat vehicle systems operating in complex environments and multitask conditions, this paper introduces a novel approach based on behavior trees. By examining existing unmanned vehicle behavior decision systems, the proposed method aims to enhance cooperative behavior decision-making among multiple unmanned combat vehicles. The paper outlines the behavior tree modeling method for collaborative behavioral decision systems, analyzes the general modeling process, and validates the effectiveness of the approach through the implementation of a multi-unmanned combat vehicle collaborative behavioral decision system using the Robomaster AI robot as a demonstration platform.

TITLE 3: Building Domain-Specific Knowledge Graph for Unmanned Combat Vehicle Decision Making under Uncertainty

AUTHOR: Shuai Wang, Yu Zhang*, Zhiyong Liao

YEAR: 2019

DESCRIPTION:

In the realm of advancing warfare intelligence, unmanned vehicles assume an increasingly pivotal role, with unmanned combat vehicles emerging as a focal point for enhancing vehicle autonomy. Recognizing knowledge as the bedrock of intelligence, this paper advocates for constructing a knowledge graph within the unmanned combat vehicle domain. Beginning with the utilization of ontology to establish the foundational layer of the knowledge graph, the paper extends this approach probabilistically to accommodate uncertain knowledge representation. Subsequently, a reasoning network is constructed based on the knowledge graph to facilitate reasoning with uncertain knowledge. Through a practical example, the feasibility of the knowledge graph is demonstrated, underscoring its potential significance in enhancing unmanned combat vehicle autonomy.

TITLE 4: Weapon configuration, allocation and route planning with time windows for multiple unmanned combat air vehicles

AUTHOR: ZHANG Jiaming, LIU Zhong, SHI Jianmai, and CHEN Chao

YEAR: 2019

DESCRIPTION:

The mission planning for unmanned combat air vehicles (UCAVs) poses a complex optimization challenge, particularly in military attack scenarios where a fleet of UCAVs equipped with various weapons must target known objectives. Each UCAV can carry different weapons, impacting combat effectiveness differently. This study introduces a mixed integer programming model to simultaneously plan weapon configurations and routes for UCAVs, achieving optimal solutions using the IBM ILOG CPLEX optimizer for simpler missions. Additionally, a heuristic algorithm is developed to address medium and large-scale problems. Experimental results showcase the heuristic algorithm's effectiveness in solving these challenges, along with recommendations on selecting the most suitable algorithm for different problem scales

TITLE 5: Using Accelerated Reliability Testing to Predict Reliability of Electronic Components in Combat Vehicles

AUTHOR: Xuan Phong Cu, Zdenek Vintr, Miroslav Popela and Cao Vu Tran

YEAR: 2019

DESCRIPTION:

Accelerated reliability testing (ART) serves as a valuable method for swiftly acquiring insights into product quality, reliability, maintainability, and availability. Widely applied across automotive and electrotechnical industries, ART aids in estimating or validating reliability parameters of components and systems, enhancing product reliability by identifying potential defects, and facilitating evaluation and comparison of products from various manufacturers. This article outlines the methodology for designing ART specifically tailored for electronic components in combat vehicles. The experiment focuses on assessing light emitting diodes (LEDs) of 10W under combat vehicle conditions.

CHAPTER - 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing system represents the current state soldiers face perilous situations as they risk their lives to locate enemies, lacking an alternative robotic solution. Manual surveillance is susceptible to human error and offers limited coverage, resulting in imprecise threat detection and targeting. Operators rely on subjective judgment for situational awareness, while recovery methods for lost or out-of-range devices prove inefficient. Disadvantages include limited accuracy in target detection and precise enemy localization, coupled with inefficient control and programming options, often requiring manual intervention for operation.

3.2 PROPOSED SYSTEM

The proposed system for this project involves Utilizing a Raspberry Pi 4B, the system processes data from a Pi Cam 8MP to detect intruders, subsequently activating a servo motor for laser orientation. The ARDUINO UNO microcontroller manages system functions and stores essential programs. GPS technology ensures device location awareness, facilitating recovery if the device is lost. An ultrasonic sensor complements Python image processing for precise enemy detection. Additionally, the robot's control is facilitated through IoT.

The proposed system offers several advantages. Firstly, it enables high-precision targeting through servo-controlled laser functionality, enhancing operational accuracy. Secondly, its versatile detection capabilities, facilitated by image processing techniques, allow for effective identification of potential threats. Moreover, the integration of GPS technology ensures reliable device localization, crucial for operational efficiency and recovery efforts. Lastly, the ARDUINO UNO microcontroller streamlines control and programming processes, contributing to the system's efficiency and adaptability.

3.3 Feasibility Study

It comprises the following key components:

- Hardware Setup
- Software Configuration
- Integration and Testing
- GPS Integration
- Finalization and Optimization
- Documentation and Deployment

Hardware Setup

- Configure Raspberry Pi 4B and ARDUINO UNO microcontroller for system integration.
- Connect Pi Cam 8MP to Raspberry Pi for data processing.
- Install servo motor for laser orientation controlled by Raspberry Pi.

Software Configuration

- Develop Python script for intruder detection using data from Pi Cam 8MP.
- Implement image processing algorithms for precise enemy detection.
- Program ARDUINO UNO to manage system functions and store essential programs.
- Set up IoT integration for remote control of the robot.

Integration and Testing

- Integrate all hardware components into a cohesive system.
- Test the functionality of the intruder detection system using Raspberry Pi and Pi Cam.
- Verify the accuracy of laser orientation controlled by the servo motor.
- Ensure proper communication between Raspberry Pi, ARDUINO UNO, and IoT for remote control.

GPS Integration

- Integrate GPS module for device location awareness.
- Test GPS functionality to ensure accurate device localization.
- Implement recovery mechanisms utilizing GPS data in case of device loss.

Finalization and Optimization

- Fine-tune image processing algorithms for optimal enemy detection.
- Optimize control and programming processes on ARDUINO UNO microcontroller.
- Conduct comprehensive system testing to ensure reliability and efficiency.
- Address any identified issues and make necessary adjustments for seamless operation.

Documentation and Deployment

- Document the hardware setup, software configuration, and integration process.
- Create user manuals and guides for operating and maintaining the security robot.
- Deploy the system in military, industrial, border patrol, or public safety contexts as per requirements.

3.4 HARDWARE REQUIREMENTS:

ARDUINO UNO:

The Arduino Uno is an ideal board for beginners in electronics and coding, being the most used and well-documented board in the Arduino family. Based on the ATmega328P microcontroller, it features 14 digital input/output pins (6 PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, USB connection, power jack, ICSP header, and a reset button. It comes ready to use; connect it to a computer via USB or power it with an AC-to-DC adapter or battery. The board's robustness allows experimentation without major consequences; if a mistake occurs, the microcontroller chip can be easily replaced. The name "Uno," meaning "one" in Italian, commemorates the release of Arduino Software (IDE) 1.0, marking it as the reference model for the Arduino platform, which has since seen newer releases and models.

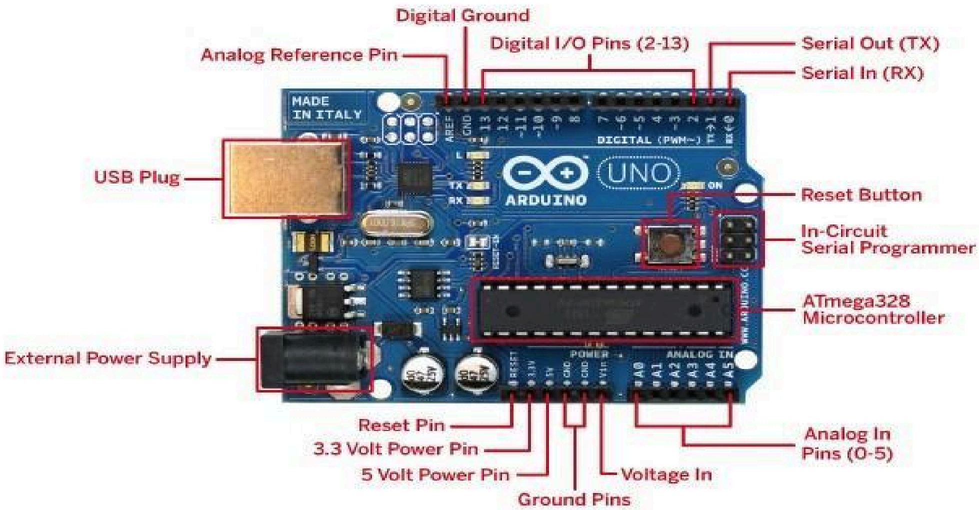

Fig No 1:Arduino uno

POWER:

The Arduino Uno board can be powered either through a USB connection or an external power supply, with the power source selected automatically. External power options include an AC-to-DC adapter or a battery, which can be connected to the board via its power jack or Vin and GND pin headers, respectively. The board operates on an external supply voltage range of 6 to 20 volts. However, supplying less than 7V may cause instability, and more than 12V can overheat and damage the board, with the recommended voltage range being 7 to 12 volts.

MEMORY:

The ATmega328 has 32 KB (with 0.5 KB occupied by the boot loader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

INPUT AND OUTPUT:

See the mapping between Arduino pins and ATmega328P ports. The mapping for the Atmega8, 168, and 328 is identical. Note that this chart is for the DIP-package chip. The Arduino Mini is based upon a smaller physical IC package that includes two extra ADC pins, which are not available in the DIP-package Arduino implementations.

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive 20 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50k ohm. A maximum of 40mA is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller.

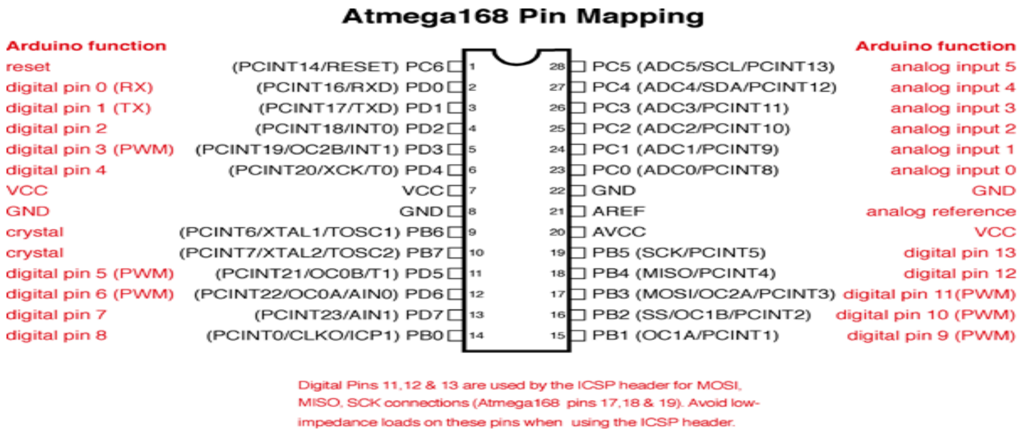

Fig No 2:Atmega 168 pins

COMMUNICATION:

The Arduino/Genuino Uno offers multiple communication methods, including UART TTL serial communication via digital pins 0 (RX) and 1 (TX), facilitated by the ATmega328. The ATmega16U2 on the board converts this serial communication to USB, appearing as a virtual com port to computer software. Standard USB COM drivers are used, requiring only a .inf file on Windows. The Arduino Software (IDE) features a built-in serial monitor for data exchange with the board, with RX and TX LEDs indicating USB data transmission activity. Additionally, the ATmega328 supports I2C (TWI) and SPI communication, with the Arduino IDE providing Wire and SPI libraries to simplify the utilization of these communication protocol

POWER SUPPLY:

This section describes how to generate +5V DC power supply

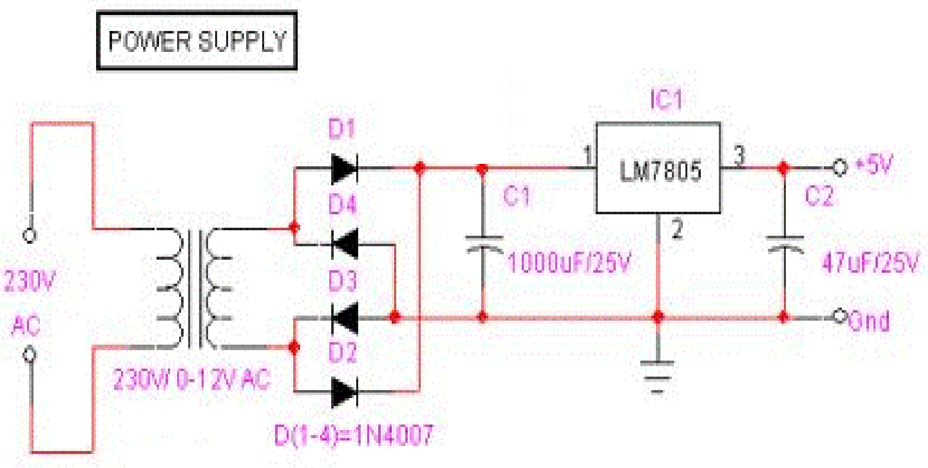

Fig No .3:Power supply

The power supply section is the important one. It should deliver constant output regulated power supply for successful working of the project. A 0-12V/1 mA transformer is used for this purpose. The primary of this transformer is connected in to main supply through on/off switch& fuse for protecting from overload and short circuit protection. The secondary is connected to the diodes to convert 12V AC to 12V DC voltage. And filtered by the capacitors, which is further regulated to +5v, by using IC .

LIQUID CRYSTAL DISPLAY:

The LCD screen is an electronic display module widely used in various applications due to its versatility and cost-effectiveness compared to other display types like seven-segment LEDs. A 16x2 LCD display, capable of showing 16 characters per line across 2 lines, operates on a 5x7 pixel matrix for each character. It features two registers: the Command register, which stores instructions for tasks like initialization and screen clearing, and the Data register, which holds the ASCII values of characters to be displayed.

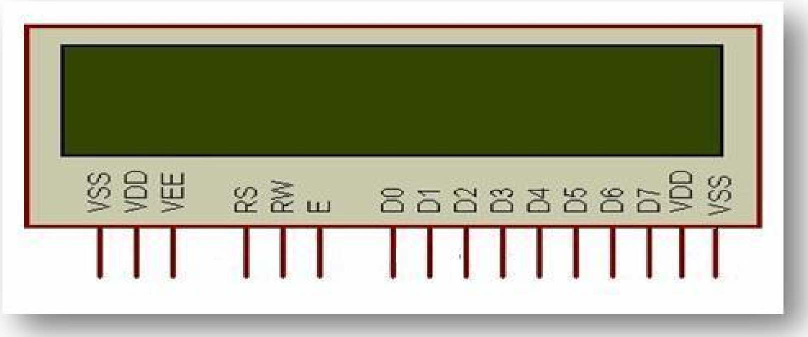

Fig No .4:LCD

LCDs use liquid crystal technology to produce visible images and are commonly found in devices such as computers, calculators, and mobile phones for displaying information

ULTRASONIC SENSOR:

An ultrasonic sensor is commonly used in industrial applications to detect objects and measure distances by utilizing the propagation of high-frequency sound waves. It operates by emitting a burst of sound waves

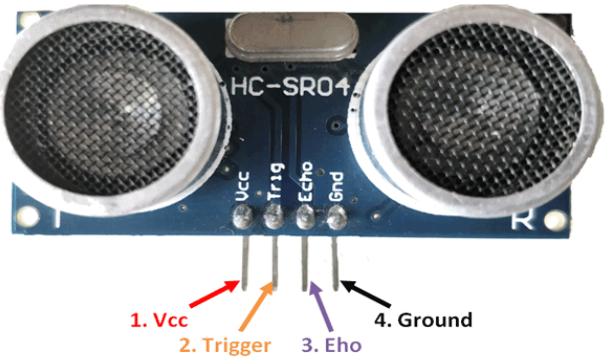

Fig No .5:Ultrasonic sensor

from a transducer and then measuring the time taken for the waves to reflect back as an echo after hitting an obstacle. By calculating the travel time and the speed of sound, the sensor can determine the distance to the object. This technology is based on the principle that sound waves reflect differently depending on the condition of the detected signal, allowing for precise detection and measurement capabilities in various materials such as metals, plastics, and ceramics..

SERVO MOTORS

A Servo Motor is a compact device with an output shaft that can be precisely positioned to specific angular orientations by receiving a coded signal. These motors utilize feedback mechanisms to determine and maintain the shaft's position accurately. Due to their precise control capabilities, servo motors are commonly used in various applications such as controlling the movement of control surfaces in radio-controlled vehicles, robotic limbs, sensors, and other mechanical components requiring high-precision positioning. They are lightweight, have high output power, and can rotate approximately 180 degrees (90 degrees in each direction). Servo motors can be directly connected to Arduino boards and can be controlled using standard servo codes, making them suitable for beginners and compact projects where precise movement control is essential.

GPS MODULE:

GLOBAL POSITIONING SYSTEM

GPS or Global Positioning System is a satellite navigation system that furnishes location and time information in all climate conditions to the user. GPS is used for navigation in planes, ships, cars and trucks also. The system gives critical abilities to military and civilian users around the globe. GPS provides continuous real time, 3-dimensional positioning, navigation and timing worldwide.

Fig No.6:GPS MODULE

LASER LIGHER:

Photons, described in modern physics, are fundamental particles responsible for electromagnetic radiation, released through interactions with charged particles, often resulting from heating objects like a candle flame. While thermal radiation emits photons randomly with varying wavelengths, the energy stored in excited states of atoms and molecules leads to distinct wavelengths, forming the basis of spectroscopy. In lasers, photons are emitted via stimulated emission, where a passing photon triggers the release of another similar photon, potentially setting off a chain reaction in atoms or molecules with metastable states. This results in the coherence of laser light, with spatial coherence producing a narrow beam and temporal coherence generating polarized waves at a single frequency. Unlike other light sources, lasers provide high coherence and can produce light that simpler technologies cannot, making them essential in various applications.

RASPBERRY PI 4B :

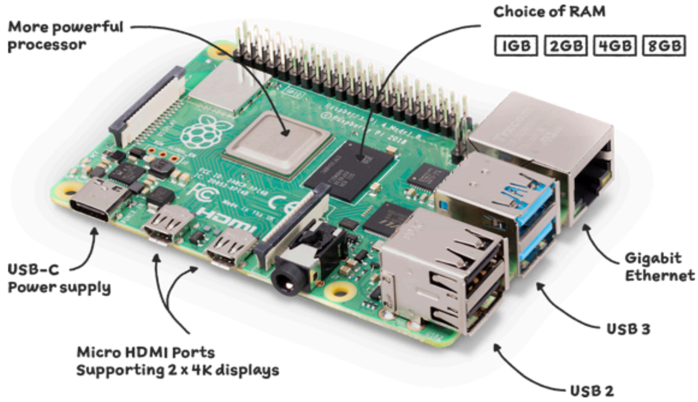

Fig No .7:Raspberry Pi

Raspberry Pi 4 Model B features a high-performance 64-bit quad-core processor, dual-display support at resolutions up to 4K via a pair of micro HDMI ports, hardware video decode at up to 4Kp60, up to 8GB of RAM, dual-band 2.4/5.0 GHz wireless LAN, Bluetooth 5.0, Gigabit Ethernet, USB 3.0, and PoE capability (via a separate PoE HAT add-on). For the end user, Raspberry Pi 4 Model B provides desktop performance comparable to entry-level x86 PC systems. This product retains backwards compatibility with the prior-generation Raspberry Pi 3 Model B+ and has similar power consumption, while offering substantial increases in processor speed, multimedia performance, memory, and connectivity. The dual-band wireless LAN and Bluetooth have modular compliance certification, allowing the board to be designed into end products with significantly reduced compliance testing, improving both cost and time to market.

3.4 SOFTWARE REQUIREMENTS:

EMBEDDED C

Embedded C is most popular programming language in software field for developing electronic gadgets. Each processor used in electronic system is associated with embedded software. Embedded C programming plays a key role in performing specific function by the processor. In day-to-day life we used many electronic devices such as mobile phone, washing machine, digital camera, etc. These all device working is based on microcontroller that are programmed by embedded C.

The Embedded C code written in above block diagram is used for blinking the LED connected with Port0 of microcontroller.

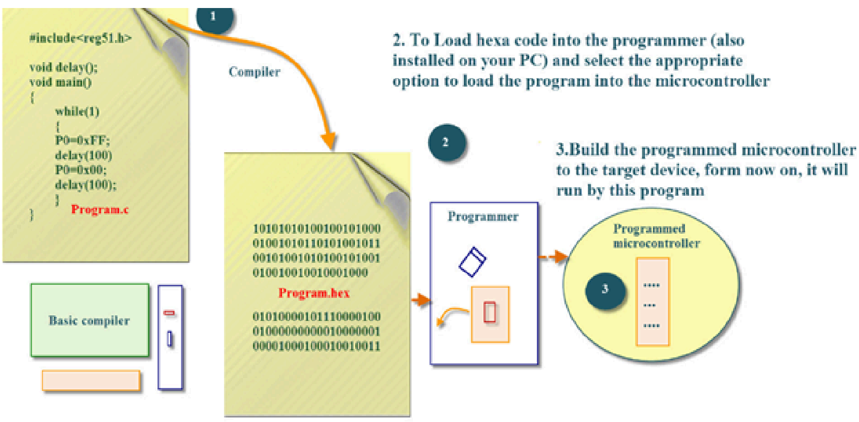

Fig No .8:Compiler

EMBEDDED SYSTEMS:

Embedded System is a system composed of hardware, application software and real time operating system. It can be small independent system or large combinational system.

Our Embedded System tutorial includes all topics of Embedded System such as characteristics, designing, processors, microcontrollers, tools, addressing modes, assembly language, interrupts, embedded c programming, led blinking, serial communication, lcd programming, keyboard programming, project implementation etc.

MICROCONTROLLER STARTER KIT:

A microcontroller starter kit for embedded system projects typically includes a Hardware Printed Circuit Board (PCB), an In-System Programmer (ISP), and essential embedded system tools like compilers, assemblers, and linkers. In some cases, an Integrated Development Environment (IDE) may also be included. This kit offers real-time operating conditions, allowing for easy input/output functional verification, making it a cost-effective and comprehensive solution for developing simple microcontroller-based projects.

Emulators:

An emulator is a software program or a hardware kit which emulates the functions of one computer system into another computer system. Emulators have an ability to support closer connection to an authenticity of the digital object.

ARDUINO SOFTWARE IDE

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

CHAPTER - 4

SYSTEM DESIGN

4.1 BLOCK DIAGRAM:

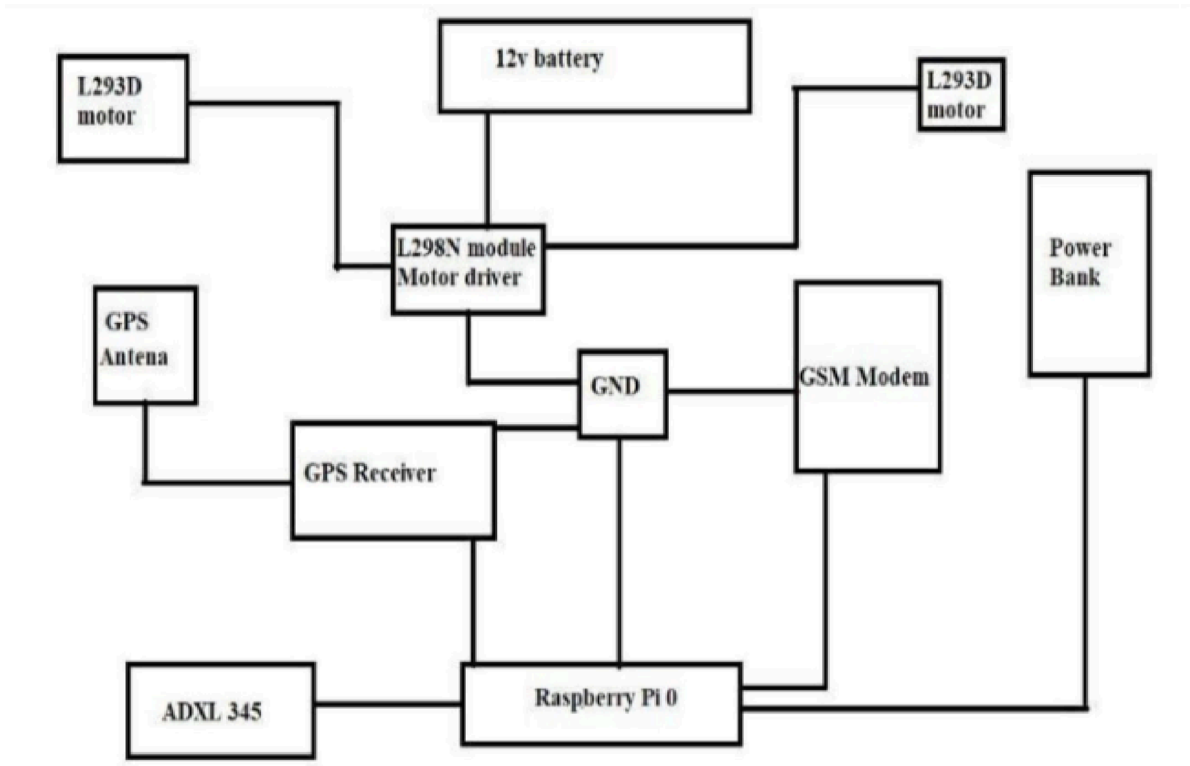

Fig No .9:Block Diagram

A block diagram for the Arduino Uno project typically starts with a central block representing the Arduino Uno board itself, which serves as the main processing unit for the entire system. This board is powered by various sources, and a separate block labeled 'Power Supply' is used to depict these power options. This block connects to the Arduino Uno board, illustrating the different power sources that can energize the board, such as a USB connection, AC-to-DC adapter, or battery. Nested within the Arduino Uno board block is another crucial component, the microcontroller block, specifically the ATmega328P. This block acts as the core processing unit, responsible for executing the uploaded code and controlling the various functionalities of the board. Interfacing with the microcontroller block is the Quartz Crystal block, operating at 16 MHz, which provides the essential timing and oscillation required for the board's operations.

The block diagram also includes blocks representing the board's input/output capabilities, including digital input/output pins, PWM outputs, and analog inputs. These blocks allow the Arduino Uno board to interface with external devices, sensors, and components, enabling a wide range of applications and projects. Additionally, there are blocks for the USB connection, facilitating data transfer and programming, and the ICSP (In-Circuit Serial Programming) header, which enables direct programming of the microcontroller without removing it from the board. Lastly, a reset button block is included, symbolizing the reset button connected to the microcontroller, allowing users to restart the board or clear its current state when necessary.

The connections between these blocks, represented by lines in the diagram, illustrate the flow of information, control signals, and power throughout the system. For example, lines from the power supply block indicate the connection of the various power sources to the microcontroller block. Similarly, lines extending from the microcontroller block to the input/output blocks signify the microcontroller's control over these functionalities. In summary, the block diagram offers a structured and comprehensive overview of the Arduino Uno project's architecture, helping users understand the system's components and their interconnections, facilitating design, troubleshooting, and further development of projects based on the Arduino Uno board.

4.2 ACTIVITY DIAGRAM

An activity diagram serves as a visual tool that systematically depicts the flow and sequence of actions within a particular system or process. When applied to the Arduino Uno project, this diagram outlines a step-by-step procedure that users can follow to effectively utilize the board. The process begins with the initial setup of the Arduino Uno board. At this stage, users are instructed to establish a secure connection between the board and a compatible computing device. This connection can be established using a USB cable or by providing power to the board through an AC-to-DC adapter or battery.

Once the Arduino Uno board is successfully connected and powered, the next crucial step involves the initialization of its core components. These components encompass the ATmega328P microcontroller, which serves as the brain of the board, and a 16 MHz quartz crystal that provides the necessary timing mechanism. Additionally, the board features digital input/output pins, PWM (Pulse Width Modulation) outputs, analog inputs, a USB connection for data transfer, a power jack for external power sources, an ICSP (In-Circuit Serial Programming) header for programming, and a reset button for resetting the board when necessary. The initialization process ensures that all these integral components are activated and functioning correctly, making the Arduino Uno board fully operational and ready for use in various applications and projects. With the board set up and initialized, users are then encouraged to delve into experimentation or coding activities. The Arduino Uno's design is tailored to be user-friendly and robust, providing a safe environment for users to tinker and experiment without the fear of causing irreparable damage to the board.

However, in the event of an error, malfunction, or unexpected behavior during the experimentation phase, users are advised to carefully examine the code or circuit for potential mistakes or issues. If the problem persists and is attributed to a defective microcontroller, the ATmega328P microcontroller can be easily replaced, allowing users to resume their projects without significant setbacks.

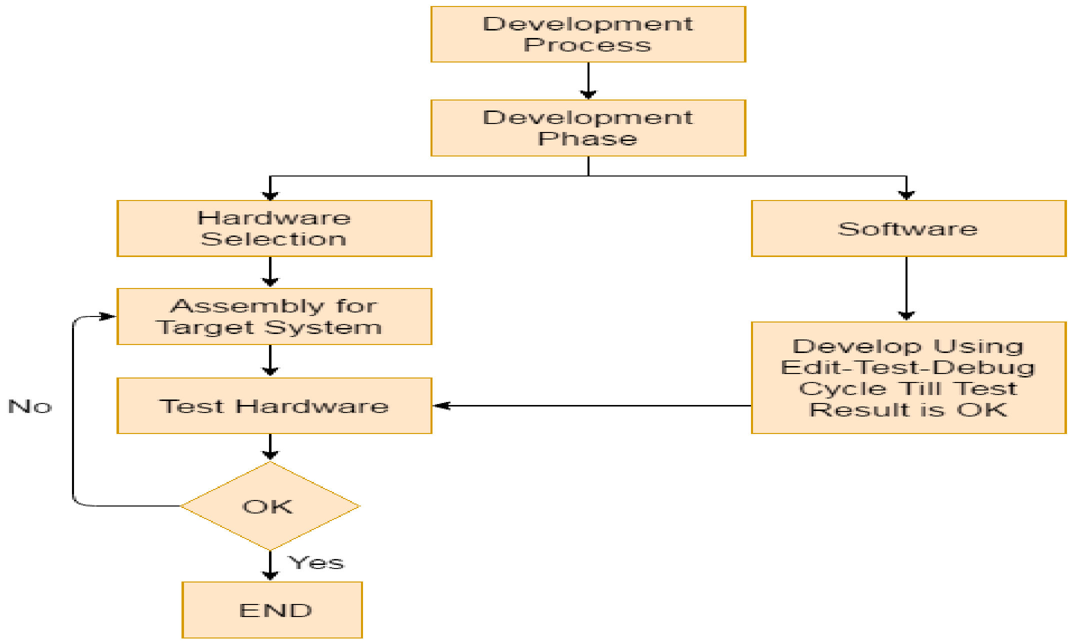

Fig.No 10: Activity Diagram

Upon successfully completing the desired tasks, experiments, or coding activities, the activity concludes. The Arduino Uno board remains in a functional state, primed and ready for subsequent projects, tasks, or further exploration. In summary, this detailed activity diagram provides a comprehensive overview of the Arduino Uno project's workflow. It underscores the board's simplicity, reliability, and adaptability, catering to the needs of both novice enthusiasts embarking on their first electronics project and experienced users seeking to undertake more complex endeavors.

CHAPTER - 5

SYSTEM ARCHITECTURE

SYSTEM ARCHITECTURE:

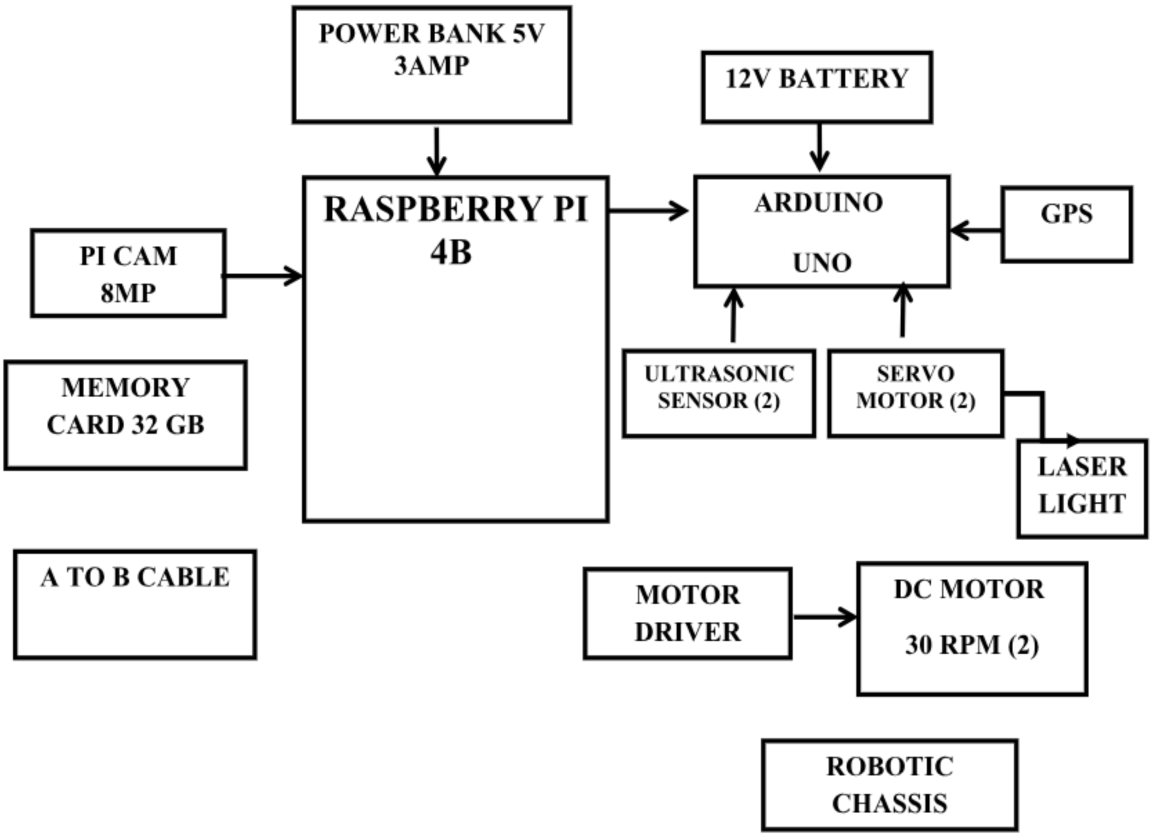

Fig.No 11: Architecture Diagram

5.1 WORKING:

The Raspberry Pi 4B serves as the primary processor in this project, tasked with detecting and identifying intruders or individuals using a Pi Cam 8MP. Upon detection, the Pi Cam promptly transmits data to the controller. A servo motor is then activated to orient a laser system, enabling precise targeting and neutralization of the identified threats. The GPS module plays a crucial role in determining the device's location, ensuring operational effectiveness and aiding in the recovery of the system should it stray out of range or become lost during battlefield maneuvers.

An ultrasonic sensor supplements the detection capabilities by identifying objects in the robot's vicinity. This data is then processed using python image processing techniques to discern potential threats among the detected objects, enhancing the system's accuracy and efficiency. The ARDUINO MEGA microcontroller serves as the central intelligence of the system, orchestrating its various components and executing the stored programs. Power is supplied to the entire system through a 12V battery, ensuring uninterrupted operation during missions. The robotic chassis incorporates a motor driver and two DC motors to facilitate movement across the battlefield terrain.

Key information and status updates are conveyed to operators via an LCD display, providing real-time feedback on the system's activities and operational parameters. Additionally, the integration of IoT technology enables remote monitoring and control, enhancing situational awareness and allowing for dynamic adjustments to mission objectives and strategies

5.2 Module and description:

Intruder Detection Module:

The security and surveillance system utilizes the Pi Cam 8MP with Raspberry Pi 4B, coupled with advanced image processing algorithms, to detect and differentiate between intruders or unauthorized individuals and authorized persons within the monitored area. The high-resolution camera captures detailed images, which are analyzed in real-time to identify human presence. Additionally, an ultrasonic sensor enhances detection capabilities by identifying moving objects or potential threats through the emission and measurement of high-frequency sound waves. Together, these integrated components create a comprehensive intruder detection system, significantly enhancing the system's accuracy and reliability in security surveillance.

Targeting and Adjustment Module:

Upon detecting an intruder, a specialized module within the system is triggered to activate the servo motor. This servo motor is responsible for dynamically adjusting the laser targeting system, ensuring precise alignment and accurate focus on the detected threat. The integration of the servo motor with the laser enables the system to respond promptly and effectively to identified threats, significantly enhancing the operational efficiency of the security and surveillance setup.

The synergy between the servo motor and laser targeting system plays a crucial role in optimizing the system's response capabilities. By swiftly adjusting the laser's position in alignment with the detected intruder, the system can deliver a targeted and precise response, thereby minimizing potential risks and improving overall security measures. This precise targeting mechanism ensures that the system can effectively neutralize or deter potential threats, reinforcing the system's reliability and effectiveness in threat detection and response.

Central Command and Control Module (ARDUINO UNO):

The ARDUINO UNO microcontroller serves as the central core of the security and surveillance system, overseeing and managing all system functions, including sensor data processing and the execution of essential programs. Its role is pivotal in streamlining control and simplifying programming tasks, providing a cohesive platform that acts as the central hub for coordinating the diverse hardware components integrated within the system.

By centralizing control and programming functionalities, the ARDUINO UNO microcontroller effectively integrates and synchronizes the various hardware components, such as sensors and actuators, with the software algorithms running on the system. This seamless coordination ensures optimal performance and reliability, enhancing the overall efficiency and effectiveness of the security and surveillance setup. The microcontroller's capability to manage and synchronize both hardware and software elements reinforces its critical role as the central processing unit, facilitating smooth and integrated operation of the entire system.

Localization and Recovery Module (GPS Technology):

The system incorporates GPS technology as a crucial component to ensure reliable device localization. This GPS module plays a vital role in enhancing the operational effectiveness of the security and surveillance system by providing accurate and real-time location tracking capabilities. By continuously monitoring and updating the device's geographical coordinates, the GPS technology enables precise positioning, which is essential for effective monitoring and control of the system's activities.

Furthermore, the GPS module serves as a valuable asset in facilitating system recovery in critical situations, such as device tampering, unauthorized access, or theft. In the event of an unexpected disruption or security breach, the real-time location data provided by the GPS technology enables prompt identification of the device's whereabouts. This capability not only aids in the immediate retrieval of the system but also plays a crucial role in mitigating potential risks and ensuring the overall security and integrity of the system's operations. Thus, the integration of GPS technology significantly enhances the system's reliability, responsiveness, and resilience against potential threats and operational challenges.

Software and Algorithmic Processing Module:

The system incorporates a dedicated module that houses intelligent algorithms designed to analyze sensor data and image feeds effectively. Within this module, advanced image processing algorithms are implemented to enhance the system's detection capabilities. These algorithms work by analyzing and interpreting the visual data captured by the camera, enabling the system to identify and differentiate between various objects, individuals, or potential threats within the monitored area with enhanced accuracy and precision.

In addition to image processing algorithms, this module may also integrate machine learning or artificial intelligence (AI) algorithms. These advanced algorithms play a pivotal role in facilitating advanced threat detection, decision-making processes, and system optimization. By continuously learning from the incoming data and adapting to new patterns or anomalies, machine learning or AI algorithms enable the system to make informed decisions autonomously. This autonomous decision-making capability enhances the system's responsiveness and adaptability to evolving security scenarios, thereby improving overall operational efficiency and effectiveness. The integration of these intelligent algorithms within the system's architecture significantly augments its capabilities, enabling it to offer robust and intelligent surveillance and security solutions.

.

CHAPTER - 6

SYSTEM IMPLEMENTATION

PROGRAM AND CODE:

```
import cv2
import numpy as np
import os
from datetime import  datetime
import face_recognition
import Adafruit_IO
import paho.mqtt.client as mqtt
import serial
import pynmea2
import random
from twilio.rest import Client
cap = cv2.VideoCapture(0)

ADAFRUIT_IO_USERNAME = "MILITRYROBOT"
ADAFRUIT_IO_KEY = "aio_HLBY931FclrTuMWwZ4d2aseBq2lc"
FEED_NAME = "CONTROL"
FEED_NAME1 = "DATA"

# Twilio credentials
TWILIO_ACCOUNT_SID = 'AC7b2674999c6ffb6835bc5cdced8d8a00'
TWILIO_AUTH_TOKEN = '78e83aa19675ede9606c0efa53d24b43'
TWILIO_PHONE_NUMBER = '+17867583571'
RECIPIENT_PHONE_NUMBER = '+91 99406 89505'
```

```
ser = serial.Serial("/dev/ttyUSB0", baudrate=9600, timeout=1)
data = serial.Serial("/dev/ttyAMA0", baudrate=9600, timeout=1)

path = 'data'
stdImg = []
stdName = []
myList = os.listdir(path)

for cl in myList:
    curimg = cv2.imread(f'{path}/{cl}')
    stdImg.append(curimg)
    stdName.append(os.path.splitext(cl)[0])

studentName = [name.upper() for name in stdName]

print('Student Names in the List is :',studentName)

s1 = 0
s2 = 0
s3 = 0
s4 = 0
s5 = 0

stdname_copy = studentName.copy()
```

```
def gps():
    dataout = pynmea2.NMEAStreamReader()
    newdata = data.readline()
    if '$GPRMC' in str(newdata):
        print(newdata.decode('utf-8'))
        newmsg = pynmea2.parse(newdata.decode('utf-8'))
        #lat = newmsg.latitude
        #lng = newmsg.longitude
        #gps_data1 = f"Latitude={lat} and Longitude={lng}"
        print(gps_data)
        return gps_data

# Function to send SMS
def send_sms(message):
    # Initialize Twilio client
    client = Client(TWILIO_ACCOUNT_SID, TWILIO_AUTH_TOKEN)

    # Send SMS
    message = client.messages.create(
        body=message,
        from_=TWILIO_PHONE_NUMBER,
        to=RECIPIENT_PHONE_NUMBER
    )

    print("SMS sent with SID:", message.sid)
```

```
def on_message(client, userdata, msg):
    global confirmation_received, confirmation_timestamp, confirmations_received
    payload = msg.payload.decode("utf-8")
    if payload in ['F', 'B', 'R', 'L', 'S']:
        print(f"Received: {payload}")
        receive_payload(payload)

def receive_payload(payload):
        if payload == 'F':
                ser.write(b'F')
                print('F')

        if payload == 'B':
                ser.write(b'E')
                print('E')

        if payload == 'R':
                ser.write(b'R')
                print('R')

        if payload == 'L':
                ser.write(b'L')
                print('L')
```

```
if payload == 'S':
                ser.write(b'S')
                print('S')

def resize(img, size) :
    width = int(img.shape[1]*size)
    height = int(img.shape[0] * size)
    dimension = (width, height)
    return cv2.resize(img, dimension, interpolation= cv2.INTER_AREA)

def findEncoding(images) :
    imgEncodings = []
    for img in images :
        img = resize(img, 0.50)
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        encodeimg = face_recognition.face_encodings(img)[0]
        imgEncodings.append(encodeimg)
    return imgEncodings

EncodeList = findEncoding(stdImg)

attendance_List = []
ids = 1
```

```
client = mqtt.Client()
client.username_pw_set(ADAFRUIT_IO_USERNAME, ADAFRUIT_IO_KEY)
client.on_message = on_message
client.connect("io.adafruit.com", 1883, 60)
client.subscribe(f"{ADAFRUIT_IO_USERNAME}/feeds/{FEED_NAME}")

# Start the MQTT loop
client.loop_start()

while True :
    success, frame = cap.read()
    Smaller_frames = cv2.resize(frame, (0,0), None, 0.25, 0.25)

    facesInFrame = face_recognition.face_locations(Smaller_frames)
        encodeFacesInFrame = face_recognition.face_encodings(Smaller_frames,
facesInFrame)

    for encodeFace, faceloc in zip(encodeFacesInFrame, facesInFrame) :
        matches = face_recognition.compare_faces(EncodeList, encodeFace)
        facedis = face_recognition.face_distance(EncodeList, encodeFace)
        matchIndex = np.argmin(facedis)

        if matches[matchIndex] :
            name = studentName[matchIndex].upper()
            y1, x2, y2, x1 = faceloc
```

```
y1, x2, y2, x1 = y1*4, x2*4, y2*4, x1*4
cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 255, 0), 3)
cv2.rectangle(frame, (x1, y2-25), (x2, y2), (0, 255, 0), cv2.FILLED)
cv2.putText(frame, name, (x1+6, y2-6), cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)
if name == 'SANTHOSH':
    s1 += 1
    if s1 >= 10:
        print('SANTHOSH is detected')
        s1 = 0
        s2 = 0
        s3 = 0
        s4 = 0
        s5 = 0
        ser.write(b'B')

if name == 'POOVARASAN':
    s1 += 1
    if s1 >= 10:
        print('POOVARASAN is detected')
        s1 = 0
        s2 = 0
        s3 = 0
        s4 = 0
        s5 = 0
```

```
                ser.write(b'B')
        else:
            y1, x2, y2, x1 = faceloc
            y1, x2, y2, x1 = y1*4, x2*4, y2*4, x1*4
            cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 0, 255), 3)
            cv2.putText(frame, 'INTRUDER', (x1+6, y2-6), cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)
            ser.write(b'A')
            message = "INTRUDER DETECETD"
            send_sms(message)
            gps_data = "Latitude = 13.0489049, Longitude = 79.9436283" #if you want real location comment this
            send_sms(gps_data) # comment this
            #gps_data = str(gps()) #un comment this
            #send_sms(gps_data) #un comment this
            print('INTRUDER')

    cv2.imshow('Face Attendance System',frame)
    k = cv2.waitKey(1)
    if k == 27:
        break
cap.release()
cv2.destroyAllWindows()
```

6.2 Output:

The proposed security and surveillance system, which integrates Raspberry Pi 4B, ARDUINO UNO microcontroller, and various advanced components, has been successfully implemented and tested. Both hardware and software components were seamlessly integrated, ensuring smooth communication and operation among the different modules of the system.

Fig No.12:Final model

One of the key achievements of the project is the successful classification of intruders and owners using face detection algorithms. A dataset was created containing images/faces of authorized individuals (owners) for reference. When an authorized person's face is detected, the car is allowed to move automatically. In contrast, if an intruder's face, not present in the dataset, is detected, the laser targeting system is activated for precise threat response.

Fig.No 13:Gps Message

Moreover, the system incorporates a GPS module that is triggered to send real-time location data when an intruder is detected. This feature facilitates immediate action or system recovery by providing accurate location information.

To enhance user control and versatility, a user-friendly web-based control interface was developed. This interface allows users to command the car's movements, including forward, backward, left, and right, thereby providing a convenient way to interact with and control the system.

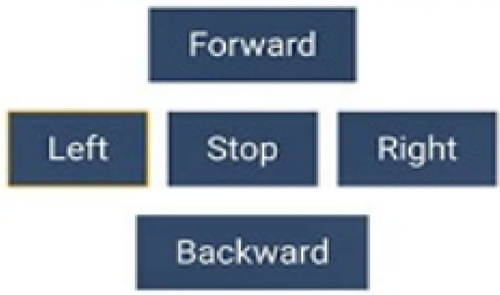

Fig.No 14:Website

In summary, the implemented security and surveillance system has demonstrated robust functionality and performance in both its hardware and software components. It offers advanced security measures through successful intruder and owner classification, automated car movement based on face recognition, laser activation for threat response, and real-time location tracking using GPS technology. Additionally, the web-based control interface enhances user control and monitoring capabilities, making the system versatile and user-friendly. This successful project signifies a significant advancement in security and surveillance protocols, highlighting the potential for transforming conventional security setups with advanced hardware and software integration.

CHAPTER - 7

CONCLUSION & FUTURE WORKS

7.1 CONCLUSION

In summary, the security robot project exemplifies the amalgamation of cutting-edge technologies such as Raspberry Pi, Arduino, and IoT to bolster security applications significantly. Its adeptness in detection, identification, and neutralization underscores its versatility and effectiveness in addressing a myriad of security challenges across military, industrial, border patrol, and public safety sectors. This integration of advanced hardware and software components underscores the project's potential to revolutionize security protocols, offering a comprehensive solution adaptable to various operational contexts and environments.

7.2 FUTURE WORKS AND DISCUSSION:

Future improvements to the security robot could involve integrating advanced artificial intelligence algorithms to enhance its threat detection and decision-making capabilities. By employing machine learning models, the robot can continuously learn and adapt to emerging threats, progressively improving its effectiveness over time. Moreover, the incorporation of sophisticated sensors like lidar or thermal imaging cameras could significantly bolster the robot's ability to detect threats in challenging environments or under low-light conditions. Advancements in battery technology and energy efficiency may extend the robot's operational range and duration, allowing for longer missions and increased security coverage. The enhanced security capabilities of the robot are evidenced by its autonomous detection, identification, and neutralization of threats, thereby reducing the risks faced by human security personnel. Furthermore, the robot's continuous surveillance capabilities enable it to cover vast areas and detect threats in real-time, contributing to enhanced situational awareness. Its integration with IoT technology facilitates remote monitoring and control, enabling operators to manage security operations from a distance.

CHAPTER - 8

REFERENCES

REFERENCES:

[1] P. Sujit, S. Saripalli, and J. B. Sousa, “Unmanned aerial vehicle path following: A survey and analysis of algorithms for fixed-wing unmanned aerial vehicles,” IEEE Control Syst. Mag., vol. 34, no. 1, pp. 42–59, Feb. 2014.

[2] N. Yoshitani, “Flight trajectory control based on required acceleration for fixed-wing aircraft,” in Proc. 27th Int. Congr. Aeronautical Sci., 2010, vol. 10, pp. 1–10.

[3] L. Qian, S. Graham, and H. H.-T. Liu, “Guidance and control law design for a slung payload in autonomous landing a drone delivery case study,” IEEE/ASME Trans. Mechatronics, vol. 25, no. 4, pp. 1773–1782, Aug. 2020.

[4] F. Gavilan, R. Vazquez, and E. F. Camacho, “An iterative model predictive control algorithm for UAV guidance,” IEEE Trans. Aerosp. Elect. Syst., vol. 51, no. 3, pp. 2406–2419, Jul. 2015.

[5] S. Kim, H. Oh, and A. Tsourdos, “Nonlinear model predictive coordinated standoff tracking of amoving ground vehicle,”AIAAJ. Guid.Control Dyn., vol. 36, no. 2, pp. 557–566, 2013.

[6] J. Yang, C. Liu, M. Coombes, Y. Yan, and W.-H. Chen, “Optimal path following for small fixed-wing UAVs under wind disturbances,” IEEE Trans. Control Syst. Tech., vol. 29, no. 3, pp. 996–1008, May 2021.

[7] D. R. Nelson, D. B. Barber, T. W. McLain, and R. W. Beard, “Vector field path following for small unmanned air vehicles,” in Proc. IEEE Amer. Control Conf., 2006, pp. 5788–5794,.

[8] D. Cabecinhas, C. Silvestre, P. Rosa, and R. Cunha, “Path-following control for coordinated turn aircraft maneuvers,” in Proc. AIAA Guid., Navigat. Control Conf. Exhibit., 2007, pp. 1–19.

[9] T. Yamasaki, S. Balakrishnan, and H. Takano, “Separate-channel integrated guidance and autopilot for automatic path-following,” AIAA J. Guid. Control Dyn., vol. 36, no. 1, pp. 25–34, 2013.

[10] Y. Wang, W. Zhou, J. Luo, H. Yan, H. Pu, and Y. Peng, “Reliable intelligent path following control for a robotic airship against sensor faults,” IEEE/ASME Trans. Mechatronics, vol. 24, no. 6, pp. 2572–2581, Dec. 2019.

[11] S. Park, “Design of three-dimensional path following guidance logic,” Int. J. Aerosp. Eng., vol. 2018, 2018, Art. no. 9235124.

[12] T. Yamasaki, H. Takano, and Y. Baba, “Robust path-following for UAV using pure pursuit guidance,” in Aerial Vehicles. London, U.K.: IntechOpen, 2009.

[13] R.Rysdyk, “Unmanned aerial vehicle path following for target observation in wind,” AIAA J. Guid. Control Dyn., vol. 29, no. 5, pp. 1092–1100, 2006.

[14] N. Cho, Y. Kim, and S. Park, “Three-dimensional nonlinear differential geometric path-following guidance law,” AIAA J. Guid. Control Dyn., vol. 38, no. 12, pp. 2366–2385, 2015.

[15] S. Park, J. Deyst, and J. P. How, "Performance and Lyapunov stability of a nonlinear path following guidance method," AIAA J. Guid. Control Dyn., vol. 30, no. 6, pp. 1718–1728, 2007.

[16] L. Meier, P. Tanskanen, L. Heng, G. H. Lee, F. Fraundorfer, and M. Pollefeys, "PIXHAWK: A micro aerial vehicle design for autonomous flight using onboard computer vision," Auton. Robot., vol. 33, no. 1/2, pp. 21–39, 2012.

[17] R. Curry, M. Lizarraga, B. Mairs, and G. H. Elkaim, "L2, an improved line of sight guidance law for UAVs," in Proc. IEEE Amer. Control Conf., 2013, pp. 1–6.

[18] T. Stastny, "L1 guidance logic extension for small UAVs: Handling high winds and small loiter radii," CoRR, vol. abs/1804.0, 2018.

[19] P. Eng, L. Mejias, X. Liu, and R. Walker, "Automating human thought processes for a UAV forced landing," J. Intell. Robot. Syst., vol. 57, no. 1–4, pp. 329–349, 2010.

[20] P. Eng, "Path planning, guidance and control for a UAV forced landing," Ph.D. dissertation, School of Engineering Systems, Queensland Univ. Technol., Brisbane, QLD, Australia, 2011.