IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

Smart Vision Enabled Low Cost Autonomous Robot

Solai Raj Muralidaran[1], Nikil Sharan Prabahar Balasubramanian[2], Niharika Elangovan[3]
Students, Department of Computer Science and Engineering
SRM Valliammai Engineering College, Kattankulathur, Tamil Nadu, India

Abstract: *Need for security and human resources for monitoring are growing nowadays. Current security systems are static in nature, lack analysis and prone to several threats and attacks. Alternative proctoring robots are hardwired, cannot reuse and expensive to implement making it unfit for Industrial and Public uses. Therefore, we propose a low-cost, re-configurable autonomous robot system for security and monitoring purposes. The robot system architecture is inspired from cloud data centre architecture where the applications are sandboxed and virtualized for efficient utilization of resources. The AI Model, Source Code, Executable scripts, internal resources are contained as a docker container. These containers are called as modules which are connected in a loosely coupled format. Modules can be replaced, added, deleted, updated, scaled over within the robot. Modules are classified as functional and auxiliary where functional modules performs AI operations, analysis and auxiliary modules performs remote results streaming, recording data also backing up footage and data to private or public cloud. The autonomous machine's camera is virtualized for simultaneous camera access by modules and to reduce computational overhead. As the resource utilization is optimized the power consumption is also reduced with combined efficiency of ARM and RISC-V chipsets. Thus, with this configurable, power efficiency, autonomous robot we hope to improve the quality of life and standards in public and industrial work places.*

Keywords: Robot, Intelligent Guided Vehicle, Virtualization, Surveillance.

I. Introduction

Theft, snatching, and law-breaking cases pose a severe threat to society, improved living, and industrial output. Despite having a range of surveillance technologies, the problems seem to be multiplying. Security systems today are static in nature, necessitating the development of dynamic, real-time, intelligent security systems. Many companies and teams have developed their own solutions to the problem, but each has its own set of issues, such as inability to reuse, hardwired setup, higher power consumption, and higher deployment cost. As a result, we devised our own solution. We were set on developing a monitoring system that used less resources, consumed less electricity, and was easier to adjust. We can maximise the usage of resources in our system with virtualisation technology, ensuring better resource utilisation. To design a system that uses less energy than current Computer Systems and Industrial Robots, ARM and RISC-V based chip sets were used. Several interconnected deep learning models offer the system's intelligence, which have been proved to produce better outcomes and be more efficient than earlier machine learning techniques and algorithms. Robot programs are loaded and executed in containers, which are self-contained sandboxes that isolate functionalities as modules. These modules are able to communicate with one another for data transfers and collaborative working. The robot's rocker boogie suspension lets it cross uneven terrain. Robot analyses surveillance video in real time, records it, encrypts it, and uploads it to a public or private cloud or saves it locally. To avoid having to set up several cameras for different duties, the robot's camera is virtualized, allowing a single camera output to be used for multiple purposes. Let's look at the methodology and outcomes of Smart vision enabled low cost autonomous robot system.

II. Literature Review

Web Controlled Raspberry Pi Robot Surveillance By V. Usha Rani; J Sridevi; P. Mohan Sai. This Paper discusses on the usage of human controlled robots for the security purposes. The robot discussed in the paper has the ability of navigating itself to any location within the range of the network and can be accessed globally from anywhere. At the core of the system lies Raspberry-pi which is small and power efficient. This research work has disadvantages such as the robot is not suited for Real time surveillance purposes, There is no Autonomous Navigation of Robot and the Robot is not intelligent enough to analyse surveillance footage on the Edge.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

Intelligent household surveillance robot By Xinyu Wu; Haitao Gong; Pei Chen; Zhong Zhi; Yangsheng Xu. This Research work specifies an indoor surveillance robot that uses audio input for analysis of situation. The robot uses Frequency coefficient to extract features from audio information. It can detect abnormal behaviours such as "falling down", "running", "crying" and "gun shooting". However, this project can only be uses on specialised indoor areas, Data present in the robot are not secured enough and does not provide data streaming. It uses older machine learning techniques for intelligence.

Design and implementation of surveillance robot for outdoor security By S Meghana; Teja V Nikhil; Raghuveer Murali; S Sanjana; R Vidhya; Khurram J Mohammed. This paper presents us a modern approach for surveillance of outdoor security. Here the robot has the ability to detect a human whether he/she is authorised or not using RFID tag carried by Persons. Wireless camera mounted on the robot provides us continuous streaming of the defined outdoor area. Cons of the Project is mainly the CPU which is Arduino UNO micro controllers which is not capable of handling large data or intelligence hence System is not intelligent enough for smart surveillance. Added to that Robot uses RFID tag data, which known to be hacked and modified easily.

Smart Surveillance Robot for Real-Time Monitoring and Control System in Environment and Industrial Applications By Anand Nayyar, Vikram Puri, Nhu Gia Nguyen, Dac Nhuong Le. This paper coins the term "Internet of Robotics" which is the field our project is trying to improve. The Project is named as InterBot 1.0. InterBot 1.0, and used for live monitoring for environmental data such as temperature, humidity and gas sensing of Industrial Machines. InterBot 1.0 is IoT based via ESP8266 which is power efficient. But this robot is not suited for Real time surveillance purposes, Interbot lacks the analytic computational power and Intelligence. Moreover, It Uses smaller micro controllers such as ESP8266.

Solar Powered Autonomous Robotic Car Using for Surveillance (2021) By V. PremchandranM. KarthikkumarV. ThamizharasanE. Sathish. This paper discusses a robotic car (unmanned vehicle) powered by solar for surveillance. The project uses Arduino UNO based micro controllers. Bluetooth voice control robot, self-balancing robot, and obstacles avoidance robot. However, the System does not contain any form of intelligent analysis method. High amount of resources needed for the implementation accounting for Low Return of Investment. It Uses smaller micro controllers such as Arduino UNO as previously mentioned which is not suitable for Edge Intelligence and Processing Surveillance Videos.

III. METHODOLOGY

The Robot it essentially performs four kinds of functionalities such as Data collection, Data Analysis, Data transfer and Navigation. The Robot consists of a built-in camera system and microphone system which collects the environmental data required for surveillance which contributes the Data collection function of Robotic system. Data analysis phase or functionality is where the data collected from camera and microphone is stored and analysed for any anomalies or violation such as a Social Detection program would analyse the collected video data for any social distance violation, the results of the data analysis program are stored in edge database and cloud as a backup and are ready to be transferred by next Phase. Data Transfer is the immediate functionality after the analysis phase, the data which are needed for surveillance such as Videos, Anomalies found in videos and robot navigation control are Streamed to Administrator's System(s) via HTTP/TCP based REST APIs.

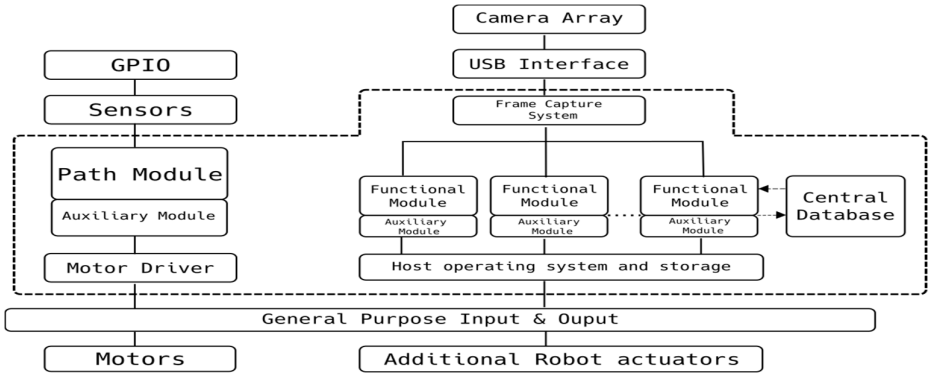

Figure 1: Robot Architecture Diagram

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

3.1 Surveillance Camera and Angle

Camera and Microphones are the major elements in collecting data for surveillance of environment. Here an high quality camera is attached to the body of the robot, the camera module is again responsible for collecting audio data as it is built in with microphones. This camera arrangement refers to the machine vision system of autonomous robot. The project aims in reducing implementation cost hence the system(robot) consists of only one camera which has is attached to a mechanical neck like system for providing 270deg of rotation around the environment. Other Robotic systems with camera contains a one to one mapping of camera to its corresponding functionality, which does increases the number camera if the functionalities are increased i.e,. If a robot wants to simultaneously perform two operations with camera video feeds it would need two cameras for separate functionality, to eliminate this difficulty we employ a frame capture system.

3.2 Frame Capture System

The Frame Capture System is a unique program that has been introduced to the robotic framework to assist with camera virtualisation and to reduce the hardware expense of adding many camera modules into the robot. The built-in camera module of the robot is interfaced with the help of Open Computer Vision packages and the Linux libusb library in the Frame Capture System. FCS (Frame Capture System) creates real-time frames from the camera stream. The FCS generates image frames that are transmitted directly to the functional modules and front end dashboard via REST APIs. A network image stream is formed that is updated on a regular basis and can be read and processed directly by the Open CV library.

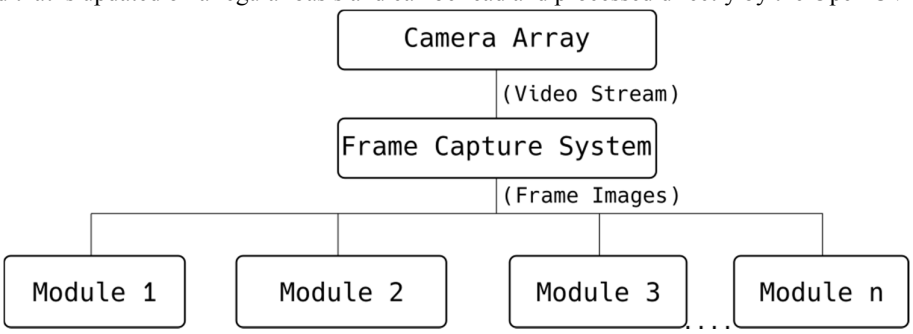

Figure 2: Frame capture system of robot

3.3 Deep Learning Techniques and Models

Artificial intelligence, is one of the most rapidly developing technologies in the world today. Its goal is to create intelligent computer systems that can think for themselves along with emulating individual's behaviour. Deep Learning is a subset of machine learning that is benefits teaching machines how to recognise or predict things, classify objects, and regress values. Deep Learning may be found in many places in today's computer world, from simple Android apps to Learning Based Content Recommendation Systems for media networks. Neural Networks are interconnected nodes with weights that can be changed according to training and error optimisations, which are commonly used in Deep Learning. The quantity of exploration, financing, and development into Deep learning modules has boosted the varieties of Neural networks during the last decade. Each neural network has its own set of properties and applications, and the Convolutional Neural Network is employed in this project for image classification and object recognition.

3.4 Convolutional Neural Network

CNN or Convolutional Neural Network is a type of deep learning Neural network model or Deep learning algorithm which is used teach a computer to classify various images and detection of objects present in images. There are a bunch of available algorithms used to classify the images and detect object in the images. But CNN gets an upper hand due to automatic pre-processing of training image set and Due to the reduced number of parameters and re-usability of weights. The convolutional Neural Network performs superior fitting to the picture data set than Feed forward Neural Networks or other Machine learning techniques for Image Classification. The Convolutional neural network or CovNet consists for three basic parts they are Convolution, Pooling and Fully Connected Layers.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

3.5 Convolution Process

The CovNet's Convolution technique is unique, which is why it's called Convolutional Neural Networks. Convolution begins with the given image, which is nothing more than a matrix or matrices of colour values. Grey scale images can be represented by a single matrix, RGB images by a Red Matrix, Green Matrix, and Blue Matrix, and CMYK colour images by Cyan, Magenta, Yellow, and Black Matrices. Matrices' values are transformed into a two-dimensional array, which is subsequently employed in the convolutional process. The kernel, also known as the filter to image, is a crucial component in this process. The Kernel is simply a square matrix with binary values of 0 and 1. The Kernel Matrix is traversed throughout the convolutional process. The image matrix and the values or kernel are multiplied together and summed for a convoluted value; this procedure is continued until the image has been traversed completely. The Convolution Process produces a convoluted image with features that are comparable to the original image but are highlighted owing to the filtering and a smaller image size than the original image.

3.6 Pooling Process

Pooling is the process of grouping or selecting values from a set of options. The Max Pooling approach is used by the majority of Convolutional Neural Network types. A square matrix for pooling size is calculated and traversed through the previously convoluted image matrix and the Maximum values discovered in the provided Pooling boundary in the Max pooling method. This method is used to reduce the input feature size and identify prevalent features from an image, lowering the computational power requirement.

3.7 You Only Look Once (YOLO)

Object detection is indeed one of the applications of artificial intelligence in which a machine or artificial intelligence model can classify and localise items in an image or video employing computer vision. Different deep learning models and architectures are used for object detection some of the notable architectures are Single shot detector (SSD), Fast and Faster R-CNN, Spatial Pyramid Pooling (SPP-Net) and You Only Look Once (YOLO). The YOLOv5 series of object detection architectures and models was built at Ultralytics open-source research facility for prospective vision AI approaches, combining lessons learned and best practices evolved through thousands of hours of study and development. The Robot's core components, which include deep learning for intelligence and smart surveillance, rely on YOLO version 5 to recognise objects accurately and quickly. Because the YOLO version 5 architecture comes pre-trained with COCO data sets involving image classes such as person, motor, cycle, car, bus, train, tie, bag pack, suitcase, and so on, which are insufficient for production level Smart surveillance, the Model is trained against a newer data set collected by us involving classes such as gun, knife, person, Face with hard hat, Face without hard hat, and so on. Each module was trained separately, resulting in unique YOLO weights for each module. Using separate models trained with different data sets had its own relevance. The YOLO version 5 architecture has a choice of pre-trained model sizes that can also be trained on bespoke data sets.
YOLO version 5 is the most recent version of the YOLONET architecture for object detection, featuring 180 percent faster processing even at frame rates of 140 frames per second, which is 180 percent faster than older object detector deep learning architectures, 88 percent smaller in size than its predecessors such as version 4 and 3, and requires less computational power, allowing nano and small-sized modules to be edge functional. YOLO Version 5 Small Sized model trained on custom data set and output as PyTorch weights file format is one of the Functional Modules of this Robot that incorporates Deep Learning modules.

3.8 Modularity and Virtualization

As Mentioned in the previous Literature review section, the robots mention the papers were hard-wired for a single use case and can be used of a sole purpose. This in fact increases the need for re-wiring, re-configuring, re-writing the core of the robot to be at least functional for newer use cases. To Eliminate these issues our robot follows a modular approach. Modules are the foundation of the robot's design; they are self-contained virtualized environments having access to the memory, computational power, graphics processing power (GPU), and even hardware devices of the host computer system. Functional modules and Auxiliary Modules are the two sorts of modules in our virtualized environment design. The AI Inference Code, the unique AI Model trained for specific purposes, and additional Data required for Operations are all contained in Functional Modules. Auxiliary modules are Data Transfer Modules or HTTP Server Modules that feed

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

Analytical Data to the robot's client system while simultaneously receiving requests from other customers. These Modules can also use General Purpose Input Output to transmit and receive information from other devices if necessary.

3.9 Functional Modules

The business logic defined by the robot operator is housed in functional modules, which are docker containers. These modules are artificial intelligence programs that make use of a machine vision system, such as a frame capturing system or a sensor array. Docker containers have direct access to the Graphical Processing Unit (GPU) and General Purpose Input/Output (GPIO) for peripherals, as well as file system mounting for data storage and retrieval. Functional modules can accept inputs and use AI algorithms to process images or data from sensors. After processing, they execute data analysis before writing the output to the host storage or sending the data to the auxiliary module for additional processing, such as streaming. “Social Distance detection Module”, “Facial Mask detection Module”, “Facial Smoking detection Module”, and “Vehicle Detection in No-Parking Areas” are some of the possible functional modules that can be created. Language-independent functional modules allow the AI software to be written in any programming language, including Go Lang, c, c++, python, or java, and to support any Artificial Intelligence and Machine Learning toolkit. YOLO-NETs and CNN-based Deep Learning Models are used in the easily available modules, which are written in Go Lang, Open CV, C++ and Python.

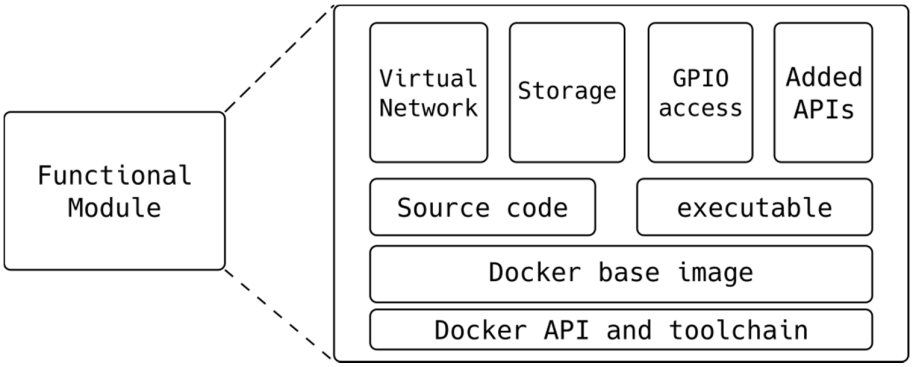

Figure 1: Functional modules of robot.

3.10 Social Distance Detection Module

- Input: Camera Video Feed.
- Output: Capturing of People who violate social distancing.
- Algorithm Used: YOLOv5 and Euclidean Distance.

The Novel coronavirus has affected the whole planet’s health, economy and mental state. An efficient way to prevent the spreading of viruses is to maintain social distance. Employing officers and workers to ensure social distancing is a difficult task and needs a lot of resources and also leaves the employees at risk. To solve this problem the module “Social Distance detection” can be installed into the Autonomous robot for automating the Social distance ensuring process.

3.11 Student Bunking Detection Module

- Input: Camera Video Feed.
- Output: Capturing People who bunk classes.
- Algorithm Used: YOLOv5 and Preloaded Timing Dataset.

In Colleges and Schools, Roaming on irregular times, bunking the classes, violating college timings, late entries and late attendance problem are a main issue. In order to maintain these parameters many colleges have installed several static and temporary solutions or even employed attenders, Teachers, professors to take care of late entries. To Solve this issue the robot can be installed with this “Unwanted Roaming Detection” module to solve this problem.

3.12 Hard Hat Detection Module

- Input: Camera Video Feed.
- Output: Capturing of People who Don’t wear safety Hard hats.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

A hard hat is a form of helmet intended to protect the head from injury caused by falling objects, impact with other objects, debris, rain, and electric shock in workplace conditions such as industrial or construction sites. Inside the helmet, suspension bands distribute the weight of the helmet and the force of any collision across the top of the head. A suspension also offers roughly 30 mm (1.2 inches) of space between the helmet's shell and the wearer's head, reducing the likelihood of an object striking the shell immediately striking the skull. A midline reinforcing ridge on some helmet shells improves impact resistance. Any employee failing to wear hard hats in industry is at danger to ensure the safety of workers. The robot can be installed with the "Hard Hat detection" Module.

3.13 Weapon Detection Module

- Input: Camera Video Feed and X-ray images from interfaced X-ray camera.
- Output: Capturing of People who possess harmful weapons.
- Algorithm Used: YOLOv5.

Carrying or possession of weapons that could bring harm to other people should be avoided and illegal in India. YoloV5 custom tweaked and trained Deep learning model helps to find the possession of weapons in a crowd or in a busy place. This information can be processed and used to raise alerts for the police stations.

3.14 Auxiliary Modules

Auxiliary modules, as their name implies, provide secondary functions that facilitate and enhance the operation of the functional module to which they are connected. A one-to-one link exists between the functional and auxiliary modules. The modules handle tasks including feeding data processed by the associated functional module into a REST API using a front end built on the ReactJS JavaScript framework and a back end built on ExpressJS and NodeJS. The following modules are also connected with an HTTP server for serving content and TLS for HTTPS security. Each module has its own port number, which is exposed on the Local Area Network and can be accessed with authentication. Data and files are also stored in host storage or the central database by auxiliary modules.

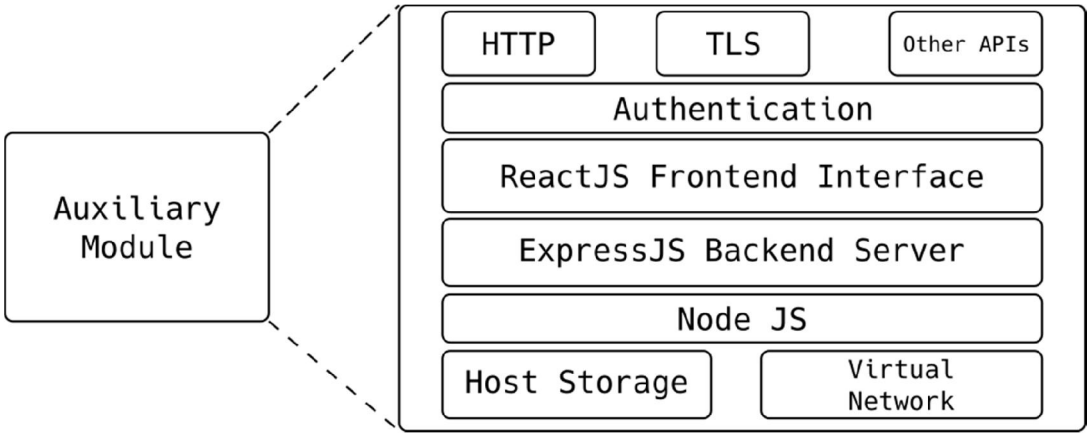

Figure 4: Auxiliary module architecture

3.15 Navigation

Modules, as previously stated, are virtual environments that enable scaling at run time. Path modules are virtualized containers that contain dependencies, executable, and source code for autonomous robot movement. With the help of a sensor array, specific search algorithms, and decision-making pipelines with respect to environmental data, the Path module obtains environmental heuristics that are used to determine a certain path and direction. Even if other modules are idle, this module is the principal module that runs on the system. The Robotic system's navigation is based on ambient data acquired by Ultrasonic proximity sensors. The collected proximity data is then fed into a rule-based navigation algorithm, which chooses the most likely safe path for the robot based on previous iterations and obstacle distances. In addition to navigation capabilities, this module has fall prevention and pit detection, with one of the proximity sensors mounted beneath the robot at a 45-degree angle to identify any pits or deeper holes in the path.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

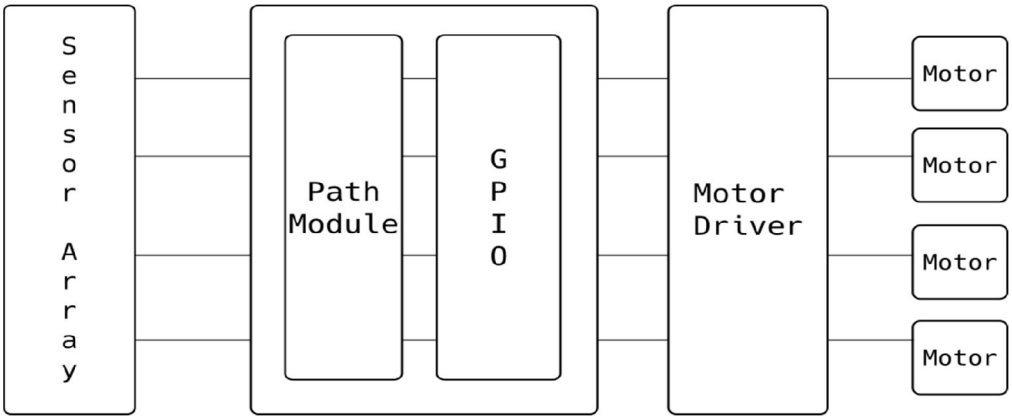

Figure 2: Navigation module or path module

3.16 Remote Navigation

In addition to the robot's autonomous navigation, the design includes a human navigation program that allows the user to manage the robot at any time and from any location. The navigation job is delegated to a secondary chip set, the ESP8266, also known as NodeMCU, by the CPU of Robot or the Raspberry Pi. The ESP8266 can run at lower voltages and has built-in Wi-Fi capabilities that can be interfaced with the CPU to enable remote navigation through the internet.

3.17 Data Collection and Analysis

Data from the camera and ultrasonic proximity sensors are collected during the robotic system's data collection phase. Frame capture system receives the camera data or visual data and virtualizes the camera input, allowing numerous modules to use a single camera. The data from the proximity sensor is sent directly to the ESP8266, which communicates with the Raspberry Pi for navigation. The system's functional modules, such as the “Social Distance Detection Module,” “Weapon Detection Module,” “Hard Hat Detection Module,” and “Student Bunk Detection Module,” receive visual data. As previously stated, visual processing takes done in the functional module using the YOLO object identification deep learning model, which is trained with a bespoke data set particular to each module. Auxiliary modules coupled with the functional modules receive the results of the functional modules.

3.18 Real time Alert System

Auxiliary module on receiving the data from functional modules they stream the data to the client system which is connected to the robot. Data sent by the functional modules also contains the Anomaly data. Anomaly data contains the Photo of the person, time and location when ever a person is suspected to violate rules provided in modules for examples, peoples who violate social distancing, carry weapons on public, fails to wear hardhats on industrial places.

3.19 Edge Storage Solution

Even if the data is live broadcast to the robot's client computer system, there is a considerable demand for recorded footage that can be used for forensic purposes afterwards. Our Robotic project also allows video footage and anomaly data to be stored on the edge. The Raspberry Pi, the robot's brain, makes it simple to connect hard drives and SD cards for extendable external and internal storage. Storing data at the edge has its own set of drawbacks, such as data loss and data theft. To get around these problems, you can store data on the cloud. Our project also includes an automated cloud backup system and LUKS encrypted edge storage.

3.20 LUKS Encryption

Linux Unified Key Setup, or LUKS, is a standard tool for disc encryption in Linux systems. TKS1 secure key setup is used by LUKS, which also provides metadata for cipher configuration management. Because our project's host operating system is Linux, LUKS is built-in, allowing for seamless encryption of external and internal data. Because LUKS is used, even if the robot's hard disc or edge storage is stolen, the data is not compromised.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

IV. RESULTS

4.1 Navigation of Robot

As previously stated, the robot's navigation is based on ambient data provided by the Robot System's ultrasonic sensors. The fully interfaced Micro controller unit with Motor driver and ultrasonic sensors required for navigation is shown in the figure 6.

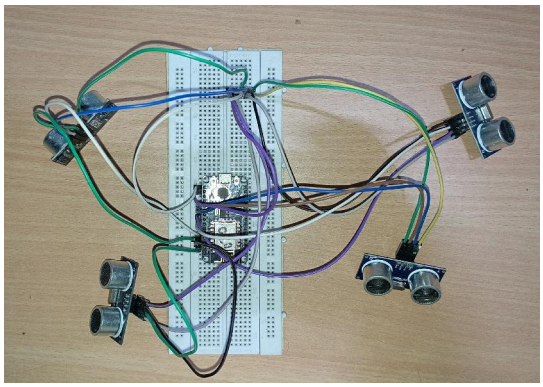

Figure 6: ESP8266 Interfaced with Ultrasonic sensors.

4.2 Functional Modules

Functional Modules are AI modules that may be altered and reprogrammed to meet specific demands and use cases. Figure 7 shows the Robot's Dashboard, which displays all currently installed functional modules. Because these functional modules rely on deep learning, they must be pre-trained before being deployed to the robot.

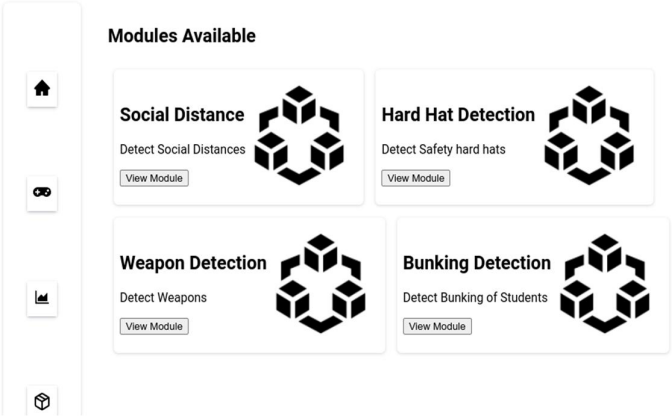

Figure 3: Dashboard showing modules installed in robot

4.3 Social Distance Detection

Social distance detection functional module should be trained before deploying. The Social detection module also has a dedicated page in the Robot's dashboard which allows users to view the images which are tagged as "Violated" or "Anomaly" by the AI model. Figure 8 displays the dashboard page with output images.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

Figure 4: Dashboard showing people violating Social Distance.

4.4 Weapon Detection

The weapon detection functional module also involves a training phase just like other modules and the figure 9 depicts the weapon detection module's training data set.

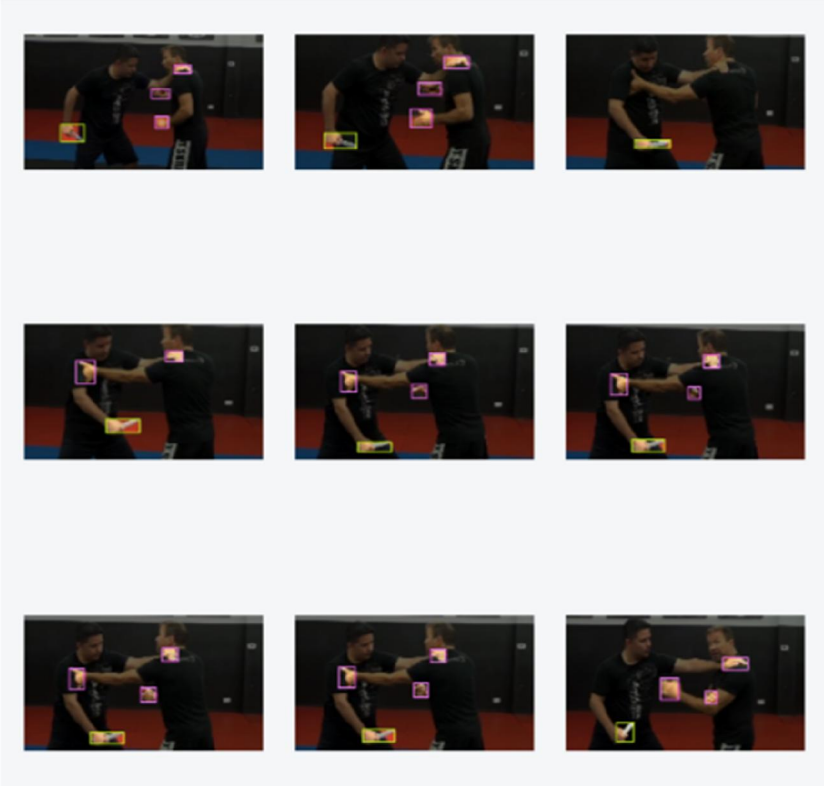

Figure 5: Weapon detection training data set.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

4.5 Hardhat Detection

Figure 11 shows the training phase of hardhat violation detection module.

Figure 6: Hard hat detection training data set.

Figure 12 shows the dashboard page with Hardhat detection module output.

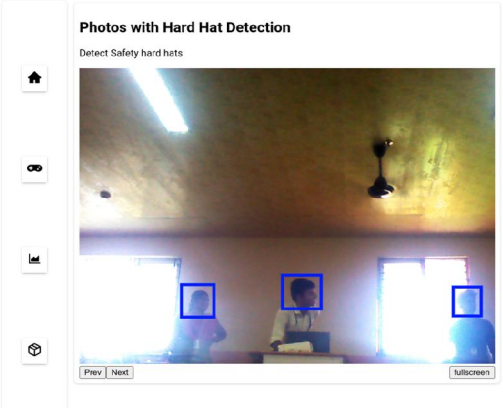

Figure 11: Dashboard with people who failed to wear hard hats.

4.6 Students Bunking Detection

Unlike other functional modules the Bunking detection module does not need a training phase as the in built data set is enough for human detection and time calculation is done for human (student) detection. Figure 13 with Students Bunking Images.

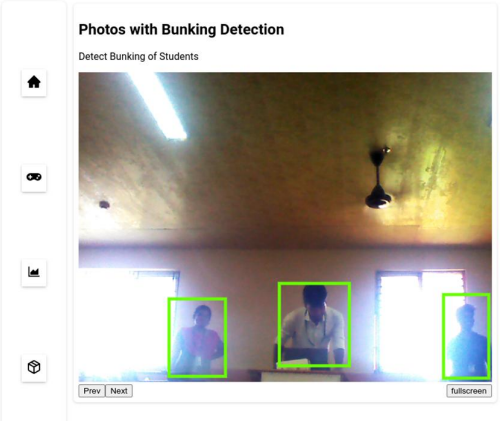

Figure 7: Students bunking detected by robot.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

4.7 Robot Dashboard

The Robot is in built with a simple and easy to use dashboard that includes a mobile web application for manual robot navigation control which is shown in figure 14.

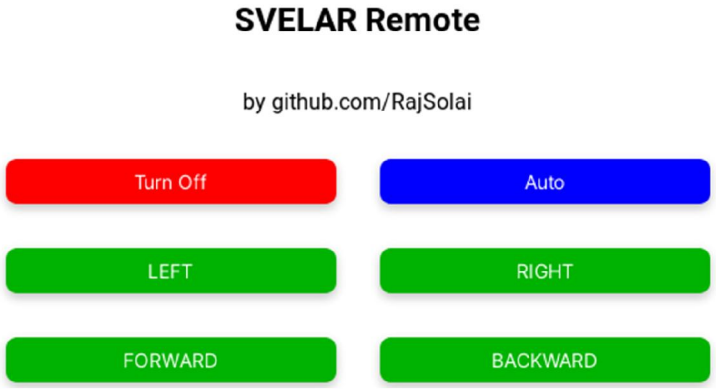

Figure 13: Robot remote control

Dashboard also work co-related with the Auxiliary modules and communicate with them using REST-API, refereed in figure 15.

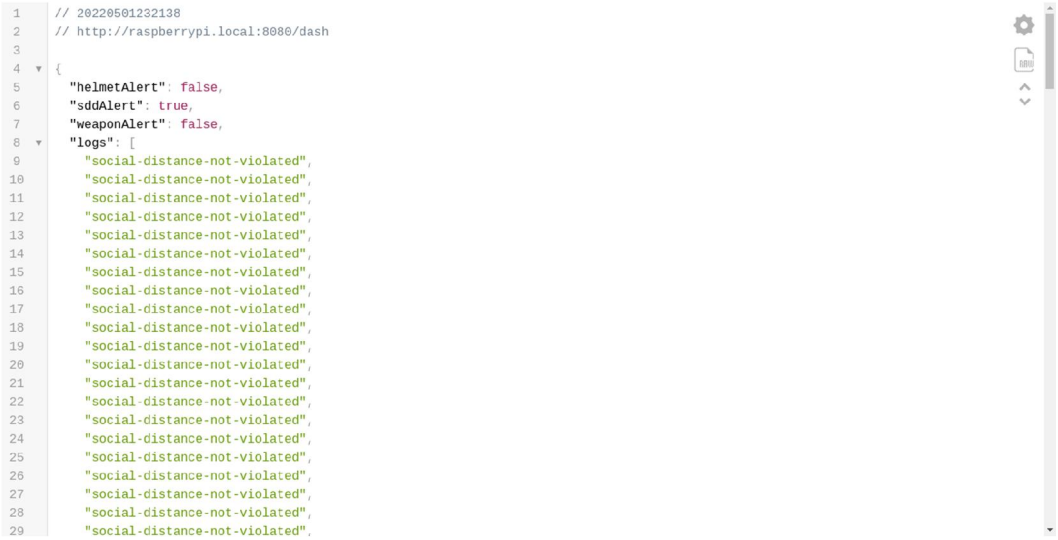

Figure 8: Auxiliary modules communicating with dashboard via REST API

Figure 15 shows the live video stream from robot to dashboard connected wireless.

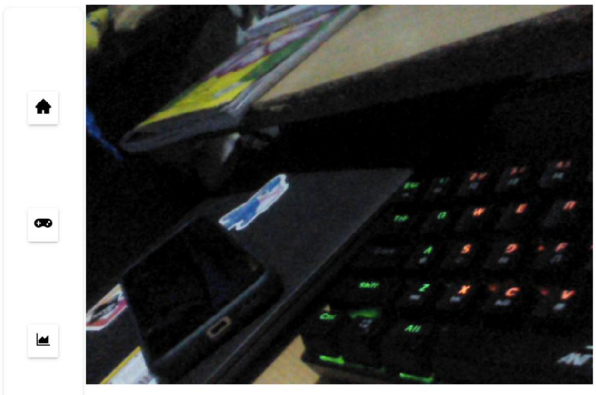

Figure 15: Live footage streaming from robot.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 3, May 2022
Impact Factor: 6.252

V. CONCLUSION

The proposed system provides Smart Surveillance based on Internet of Robotics. It can be deployed in closed environments which are considered dangerous for humans and can navigate through the environment while providing surveillance. Depending on the functional modules configured, the robot can perform a variety of tasks. The system is equipped with edge storage and thus is able to provide lossless surveillance data with a fast retrieval time. The collected data can be converted into graphs and charts for easy analysis which can be accessed through the dashboard. Since the dashboard is a web application it can be accessed using either a mobile device or a computer making it highly accessible. The system is cost and power efficient and is tremendously cheap compared to existing solutions and is thus accessible even by small scale industries or individuals unlike existing solutions which can only be used by huge corporations.

REFERENCES

[1]. S. M. Metev and V. P. Veiko, Laser Assisted Microtechnology, 2nd ed., R. M. Osgood, Jr., Ed. Berlin, Germany: Springer-Verlag, 1998.

[2]. H. Stavelin, A. Rasheed, O. San, and A. J. Hestnes, Marine life through You Only Look Once's perspective. arXiv, 2020. doi: 10.48550/ARXIV.2003.00836.

[3]. C. Diaz Alvarenga, N. Basilico, and S. Carpin, "Delayed and Time-Variant Patrolling Strategies against Attackers with Local Observation Capabilities," in Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, 2019, pp. 1928–1930.

[4]. V. Premchandran, M. Karthikkumar, V. Thamizharasan, and E. Sathish, "Solar Powered Autonomous Robotic Car Using for Surveillance," in Intelligent Manufacturing and Energy Sustainability, 2022, pp. 249–256.

[5]. S. Meghana, T. V. Nikhil, R. Murali, S. Sanjana, R. Vidhya and K. J. Mohammed, "Design and implementation of surveillance robot for outdoor security," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, pp. 1679-1682, doi: 10.1109/RTEICT.2017.8256885.

[6]. Nayyar, A., Puri, V., Nguyen, N.G., Le, D.N. (2018). Smart Surveillance Robot for Real-Time Monitoring and Control System in Environment and Industrial Applications. In: Bhateja, V., Nguyen, B., Nguyen, N., Satapathy, S., Le, DN. (eds) Information Systems Design and Intelligent Applications. Advances in Intelligent Systems and Computing, vol 672. Springer, Singapore. https://doi.org/10.1007/978-981-10-7512-4_23

[7]. Gaikwad, B., Karmakar, A. Smart surveillance system for real-time multi-person multi-camera tracking at the edge. J Real-Time Image Proc 18, 1993–2007 (2021). https://doi.org/10.1007/s11554-020-01066-8

[8]. H. R. Everett, A. M. Flynn, "A Programmable Near-Infrared Proximity Detector For Robot Navigation," Proc. SPIE 0727, Mobile Robots I, (25 February 1987); https://doi.org/10.1117/12.937800

[9]. Xinyu Wu, Haitao Gong, Pei Chen, Zhong Zhi and Yangsheng Xu, "Intelligent household surveillance robot," 2008 IEEE International Conference on Robotics and Biomimetics, 2009, pp. 1734-1739, doi: 10.1109/ROBIO.2009.4913263.

[10]. G. Jocher et al., ultralytics/yolov5: v6.1 - TensorRT, TensorFlow Edge TPU and OpenVINO Export and Inference. Zenodo, 2022. doi: 10.5281/zenodo.6222936.

[11]. Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens,Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas,Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

BIOGRAPHY

I'm Solai Raj M from Chennai, India perusing my Bachelor's of Computer Science and Engineering in SRM Valliammai Engineering College. I develop Mobile, Linux applications including IoT and automation projects. Currently, am working on better usability on Linux devices and seamless home automation with robots.