IMPLEMENTATION OF SMART AUTONOMOUS SURVEILLANCE ROBOTIC VEHICLE

[1]Karthik Hebbar M, [2]Skandapriya B N, [3]Suhas Amarnath, [4]Tejas C B, [5]Nagesh Kumar D N

[1,2,3,4]Student, Department of ECE, Jyothy Institute of Technology, Bengaluru-560082

[5]Assistant Professor, Department of ECE, Jyothy Institute of Technology, Bengaluru-560082

Abstract: Prevention of trespassers and other potential threats in unauthorized buildings, industrial areas and similar restricted places have become a tedious task. Therefore, this paper provides a viable solution to the usage of conventional surveillance of CCTV. Surveillance is the observation of behavior, activities or information for the purpose of gathering data, influencing, managing or directing. The conventional method needs human monitoring which can lead to loss of data and might cause errors. This can be avoided by designing a smart autonomous surveillance robot using artificial intelligence and IoT. This paper emphasizes on the autonomous localization of the robot using the LIDAR sensor which is achieved through incorporating SLAM algorithm and the robot's capability of distinguishing the known and unknown person's face when he/she enters the patrolling area using face recognition algorithm. Additionally, an app is designed in such a way that, whenever the intrusion occurs, it will intimate the user by sending a captured image of the intruder's face.

Keywords **- Surveillance and Security System, Autonomous, Robot, LiDAR, IoT, KNN Algorithm, SLAM.**

I. INTRODUCTION

Surveillance bots are designed and built to meet specific needs. Some areas need to be placed under video surveillance, and therefore equipping them with stationary CCTV systems is not viable. The vigilance of surveillance bot never falters, even when performing the most repetitive tasks. Research in robot mobility has had a breakthrough in the recent years, wherein, the robots have the ability to navigate in complex surroundings, patrol, safeguard, and protect as human guards would do. With the provision of their own cameras and sensors, they can also be used as mobile surveillance nodes augmenting a network of stationary-situated cameras. In this particular project we have come up with a solution to tackle surveillance errors caused by the humans. This surveillance robot is equipped with LiDAR sensor for the autonomous navigation along with an option for manual override. The LiDAR sensor gives out light waves in 360 degrees and the light which is bounced back from the objects is used by the sensor to track its path and the robot moves accordingly.[1]

YDLIDAR X4 is formulated on laser triangulation principle of ranging and uses high-speed vision acquisition and processing hardware. The core of YDLIDAR X4 runs clockwise to execute a 360-degree omnidirectional laser range scanning for its surrounding environment and then generate an outline map for the environment. The LiDAR is connected to the Raspberry Pi model 4 for the processing of the data it receives from the surroundings.

II. LITERATURE REVIEW

Face recognition plays a very important role in security systems. There are several algorithms that can be followed to achieve face recognition and this paper which is written by Eko Setiawan et al. talks about one such algorithm that is K-Nearest Neighbors. Taking an example of smart buildings which was one of the top issues, face recognition plays a significant role in the security system. The security system includes face recognition which makes use of several techniques like change in expressions, pose variations etc., Comparing various algorithms of face recognition techniques like PCA, LDA and KNN, KNN seems to provide the best results. K-Nearest Neighbor or KNN is a method of data classification where each pixel in face gives unique information. The proposed algorithm makes use of the pixel matrix to reshape in vector before classification and all the required calculation is carried out. This was implemented using OpenCV library in C language. The main processing unit used was Raspberry Pi equipped with ARM11 700 MHz core. The picture was being captured continuously from a webcam periodically and the image captured was pre-processed by applying face-detection feature. Haar-cascade was applied because of its sturdiness in face-detection. The pictures were captured continuously and the process was repeated to produce a video-based face recognition. KNN algorithm gave best accuracy with lower processing time.[2]

The paper written by Viska Mutiawani et al. talks about the alerting system developed on an Android Application. This application was developed to monitor and track vehicles. It notifies the user about the movement of the vehicle from one point to another by making use of GPS and Google Maps. The application was built on Agile software methodology. Several test cases were considered and experiment was conducted to know the accuracy results. GPS was used to locate the exact location of the vehicle; GSM module was used to communicate with the vehicle and RFID was combined for additional security. The application consists the latitude, longitude and altitude data which are the location coordinates. This data is being transmitted continuously to the cloud database. The Android application is built using Arduino Studio which consists of features like location sharing, real-time data and notification. Google firebase was the cloud storage that was used for storing the data. Several tests were carried out to measure the accuracy by comparing the results from the application and manual distance.[3]

This paper written by Muhammad Rivai et al. is on 2D Mapping mobile robot using LiDAR which was implemented to map a room in robot environment. LiDAR stand for Light Detection and Ranging which is a sensor which can be used to detect obstacles and generate a virtual map. Raspberry Pi 3B was equipped with LiDAR which sends data wirelessly. The communication between computer and the robot was done using robot operating system (ROS). To locate the exact location of the robot in the environment, Hector SLAM algorithm was used which matches the LiDAR data. The LiDAR gives data about the obstacles present around it. By using this data, the robot can construct its map automatically and move around avoiding obstacles. This can detect both dynamic and static obstacles and keeps scanning every second so that the obstacle avoidance is fast. ROS is an open source that is used to produce mapping visualizations. Usage of LiDAR plays a vital role in intelligent moving robots. The YDLiDAR X4 sensor, an indoor 2D scanner was used for mapping the room. ROS and SBC. Measurements were taken by carrying out experiments of various cases and results were noted down.[4]

This paper talks about an indoor robot that can move autonomously and can detect changes in the surrounding environment. This kind of robot can be used for security purposes in offices and by house owners. This robot is being developed using Raspberry Pi development board which runs on Raspbian OS and Robot Operating System. It moves around autonomously in the instructed routes on predefined schedule and collects the sensor data simultaneously. Here, the main server behaves as the ROS master and mobile robot behaves as a ROS slave. It consists of various functionalities like night vision, human recognition, indoor mapping and autonomous navigation. The movement of the robot is carried out using 2 geared DC motors which operates at 12V and runs at 5000 RPM. A L298N motor controller is connected to Raspberry Pi using GPIO pins. A NoIR camera is being used to take the visual input. Indoor map is created by using Lidar data which uses Hector Slam algorithm. Face recognition is performed making use of OpenCV, Python and Deep Learning method. The robot is tested under various conditions and is not affected by dark environments. Arduino-Mega was used for increasing memory to execute all the codes under one system.[5]

III. PROPOSED SYSTEM

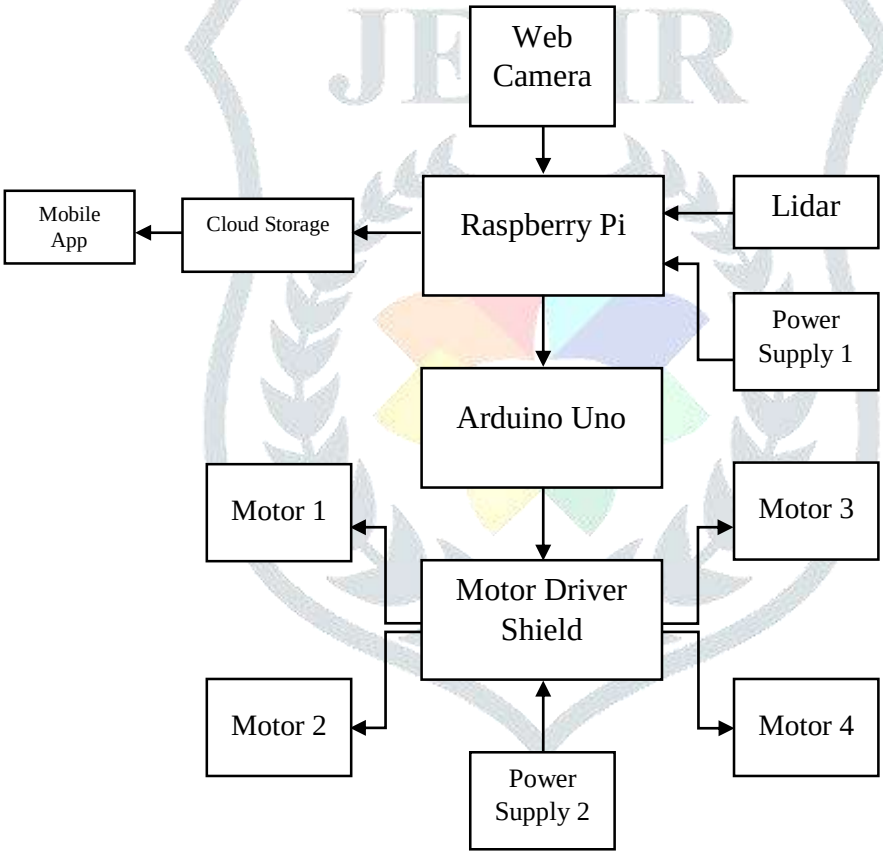

Figure 1: Block diagram of the robot

This proposed surveillance robot decreases errors caused due to human monitoring which may result in loss of critical data. The robot has been developed to prevent intrusion and potential threats in unauthorized locations. Figure 1 represents the block diagram of the proposed surveillance robot. It is implemented using hardware components like Raspberry pi, Arduino uno, motor driver shield, motors, camera and is equipped with two power supplies. Raspberry pi and Arduino uno works in master slave combination. The raspberry pi controls/sends signal to arduino uno through a USB cable using ROS serial communication. For the autonomous movement of the robot, it uses lidar sensor. The amount of time taken for the light which hits an object or a surface and then reflects back to the scanner is being calculated by the sensor. This results in creation of a map of the surroundings at regular intervals of time. The robot then navigates in a user defined fixed path with the help of the map generated by Lidar. Then the fully processed data is sent to slave (arduino uno) which in turn is connected to motor driver shield. All the motors are driven through the shield. It is equipped with four gear motors, each of them connected to a wheel. A web camera of 1080 pixels is connected to raspberry pi so as to detect the faces of intruders using face recognition algorithm. A database of all the allowed or known personnel is created in order to perform facial recognition. If the face doesn't match with the pre-fed dataset, the robot immediately alerts the admin by sending the picture of the intruder's face through an android application. The face recognition process is done through raspberry pi

and the data is sent to cloud storage using google firebase. The same is retrieved through the android application. The admin can act upon the inputs given by the robot. Two separate power supplies have been connected to the robot. One is connected to the lidar sensor and raspberry pi board and another is affixed to the motor driver shield. The reason behind connecting two power supplies is that the raspberry pi board requires uninterrupted current supply.

A. Image Processing

K Nearest Neighbor is one of the simplest image classification algorithms in machine learning. Here 'K' refers to closest samples to the image. This algorithm has an accuracy of 92% - 97%, also the accuracy depends on the number of images used in the dataset. To give a brief about the KNN algorithm, imagine 2 categories A and B respectively. Category A contains the dataset of person A who has certain facial qualities and category B contains the dataset of person B who has facial qualities different form person B. Now, in order to categorize, an input image has to be fed. An image is taken as an input and the KNN algorithm is applied to it. The facial features of the input image are compared to the dataset images of category A as well as category B. Certain features such as distance between the eyes, the structure of nose etc. are compared. After the facial features are compared, whichever category matches the input image, the input image will be classified to that category.

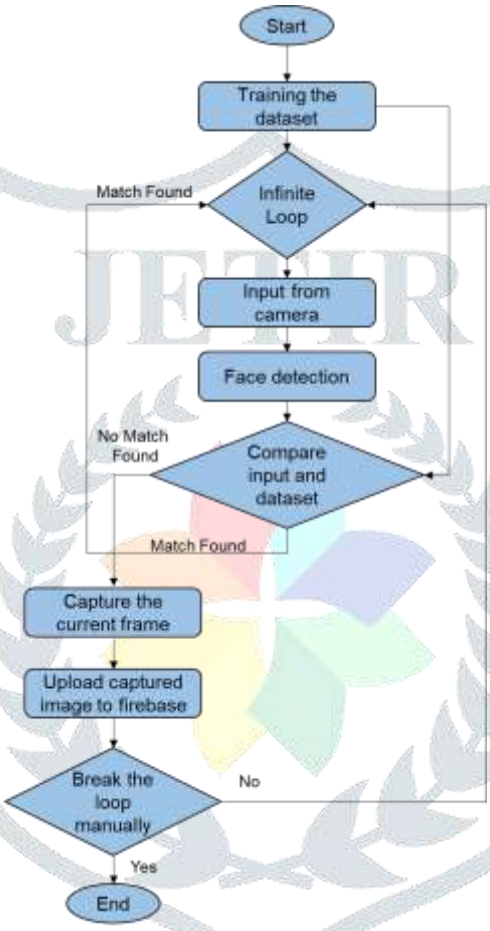

Figure 2: KNN Algorithm's Flowchart

First and the foremost, the input data, in this case is the frames which contain the known images of people are given for training the images. The above figure 2 shows the flowchart of KNN algorithm. This process of training is also known as training the dataset. Once the images are trained, the KNN algorithm works in the following way. The camera acts as an input device for the algorithm in order to classify the known from the unknown faces. Whenever a face is detected, the algorithm compares the face in the camera to the images in the training dataset. If the images from the camera and the dataset matches, then the process goes through an infinite loop till the camera finds an unknown face. If a match is not found, the camera captures the image of the face in the frame and uploads the image to Google firebase. The uploaded image can be viewed by the user through the developed android. The loop is broken manually through the android application and the whole process repeats again.

B. Alerting System

The alerting system is embedded in this project by developing an android application which alerts the user in case of intrusion. The application is developed with the help of Android studio software. Android Studio is an Integrated Development Environment (IDE) for Android app development. In our case, the application is built with JAVA as a programming language. The alerting system is crucial in case of intrusions and hence the user can take necessary actions in order to protect data.

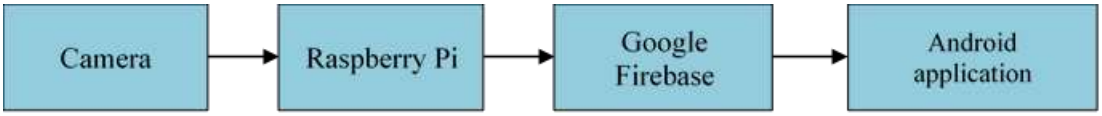

Figure 3: Block diagram of alerting system

The block diagram of an alerting system is represented in the above figure 3. The web camera which has 1080 pixels of resolution and captures 30 frames per second detects the unknown entity while patrolling, captures the face and uploads the image to the cloud database in real time through raspberry pi. We have used Firebase to store the data in cloud. Firebase stores data is an unstructured manner in JSON- like documents. JSON is a file format for storing data. This firebase platform acts as Backend-as-a-Service solution for mobile applications, involving services for building, testing etc., The captured image hence is stored in firebase storage in 'jpg' or 'png' file format. Different parameters or the aspects of the uploaded image are stored in firebase realtime database. These aspects/parameters include date, time and url of the captured image. This data is retrieved through the developed android application. The application contains login page for admission as a reason for better security. It has features like alerts in it. The user is able to view the image of the present and past intruders including the parameters like date and time of the captured image. Hence with the help of this, user will be able to carry out further instructions.

C. Autonomous Localization

Localization is a stage which is implemented in the majority of robots to locate with an insignificant margin of error. If the decision is made independently on the intelligence provided into the system, it is known as Autonomous Localization. Autonomous Localization can be achieved by using a Lidar sensor. Lidar works by determining the range which is variable distance by targeting an object by laser and correspondingly measuring the time for the reflected light to bounce back to the receiver. This sensor has capability to make 3D representations and help the robot move accordingly. With the help of autonomous localization, the robot can move around in locations with obstructions like human movement and other static obstacles. The Lidar sensor can detect the obstruction and relies on several algorithms to form its different route to reach its predefined destination. Various algorithms can be implemented into the robots according to the applications.

Figure 4: Autonomous localization process

The proposed system uses Robotic Operating System. ROS is not an actual operating system; it is a software package which consists of thousands of packages. In our case, several packages will be used to achieve autonomous localization. One of such packages is SLAM package. The Simultaneous localization and mapping involve several processes as shown in the figure 4. A SLAM algorithm takes Lidar data as input and starts producing a map of the environment. For this purpose, Hector Slam algorithm is used. The robot has to be controlled manually through a workstation computer which is connected to the robot through ROS in order to create the map. Both robot and the workstation computer are connected to the same network through Wi-Fi. The map generated is a 2D map and can be visualized using a tool called Rviz. It is a 3D visualization tool package under ROS. It is used to show certain set of information from lasers, cameras and many more sensors. Once an initial map is ready, the algorithm uses this map to simultaneously locate the robot's position and orientation in the generated map.

Once mapping and localization part is over, the robot relies on 'move Base' node of ROS in order to create a path and autonomously navigate from one point to another by avoiding all the static and dynamic obstacles. We have to set certain parameters of the robot-like dimensions of the robot, inflation radius, scanning and publish frequency. With data given to the 'move Base' node, it'll be able to generate a path to reach the destination. This process is referred to as Path Planning. This node is also responsible for the

movement of the robot i.e., after path planning it controls the velocity and direction of the motors in order to reach the destination without colliding with any obstacles.

IV. Results And Discussion

Figure 5 illustrates the recognition of known face using the KNN algorithm and is displayed using Pycharm IDE. Here a dataset of the known person's face was created and trained using the algorithm. When the person came in the vicinity of the camera it captured the image and recognized it with the person's name as the model was trained. More the images fed to the model greater will be the accuracy of the result. In our instance we have trained the dataset with 8 images for better accuracy.

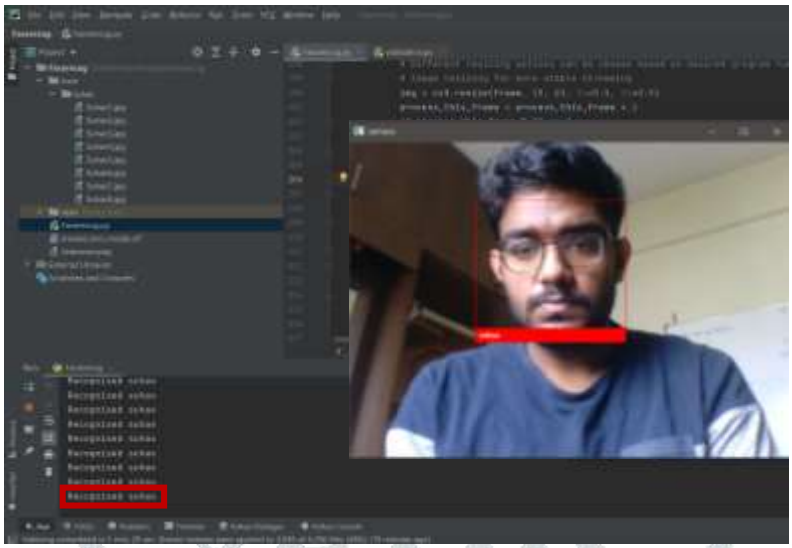

Figure 5: Recognition of known face

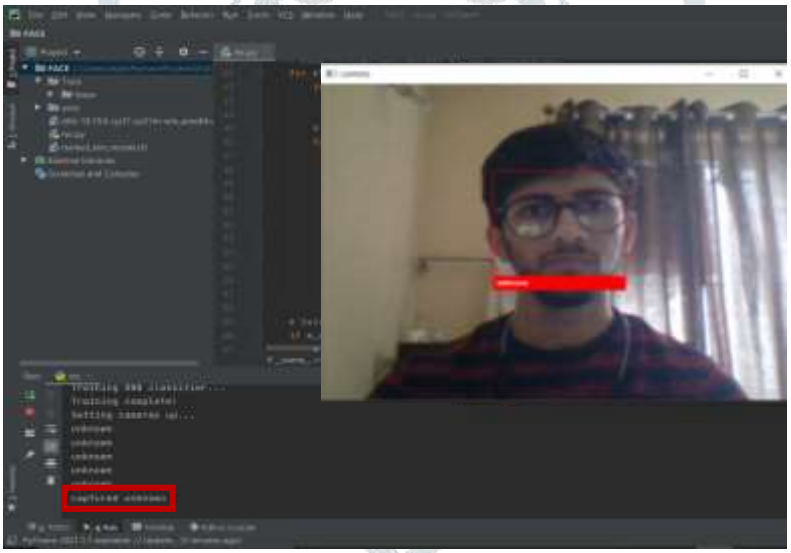

Figure 6: Recognition of unknown face

Figure 6 depicts the recognition of unknown face using the KNN algorithm and is displayed using Pycharm IDE. Here some other dataset of images were fed. When the person arrived in the region of the camera it recognized him as unkown and captured the image since it didn't match with any of the pre-fed dataset and alerted the user using developed android application.

Figure 7: Alert page of the application

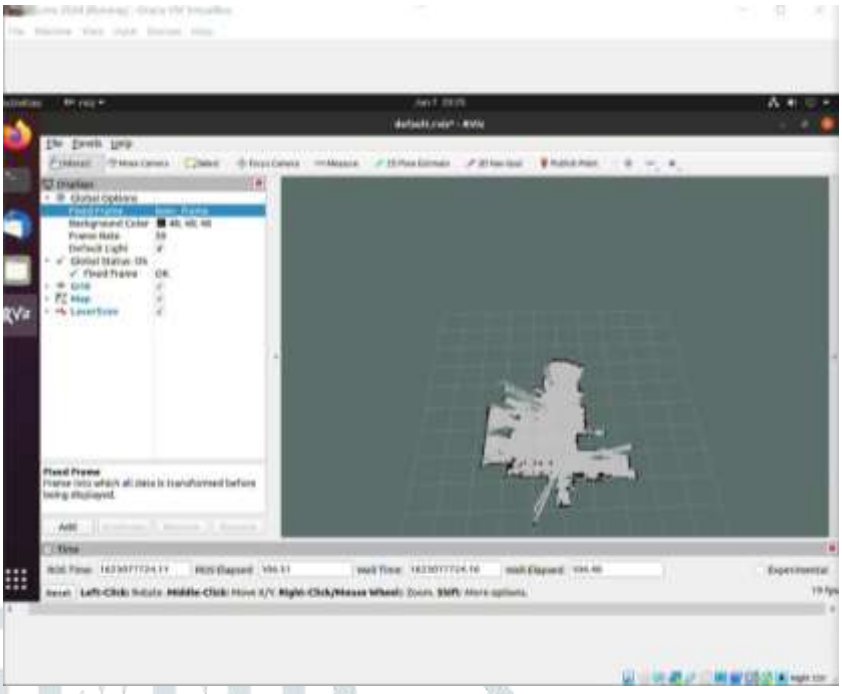

Figure 8: Map generated using hector slam

An android application was developed to retrive data from the firebase such as image and other earlier mentioned parameters. The IoT system used a raspberry pi board for the processing of the data and the same was sent to the firebase. This data was retrived using the developed android application. Figure 7 is the alert page of the application where it displays the image of the unknown people with the additional parameters like date and time of the image captured. A login page of the application is also created which provides access only to the authentic user and the data can be seen once the login is done.

Figure 8 represents the map generated using hector slam algorithm. This was created when the robot was controlled manually for the purpose of the genration of base map. This was visualized using the Rviz tool in the workstation computer. The black part in the map represents the obstacle present in the locality of the robot and the grey part is the free space for the navigation purpose.

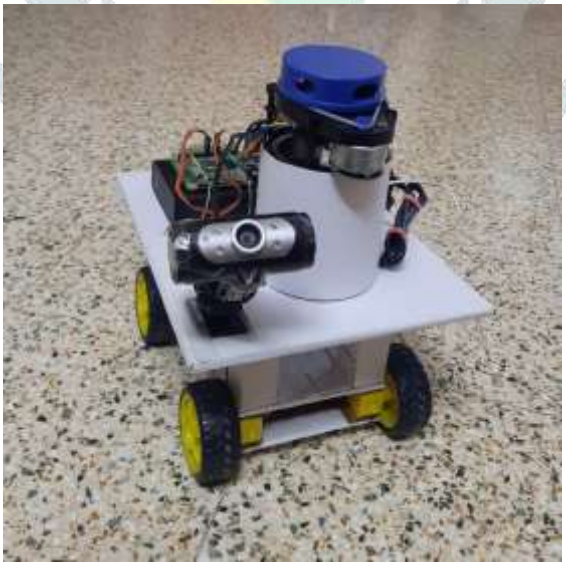

Figure 9: Working prototype of the robot

To implement the above mentioned functionalities a working prototype of the robot was developed as shown in figure 9. The robot has been designed in a multilayered structure. The base layer of robot consists of motors which are in turn connected to the wheels for the movement of the vehicle. The middle layer consists of arudino uno and motor driver shield with a power source. The topmost layer consists of camera, raspberrypi and lidar with another power source. This proposed design is feasible to accommodate the required components.

V. Conclusion

Surveillance plays a very important role in protecting data. Data may be of any kind and utmost care is taken in order to safe guard these data. This particular paper, an efficient and reliable surveillance bot is developed which can autonomously move with the help of LIDAR technology and detect intrusions using face recognition algorithm. KNN algorithm is used for facial recognition as it has a better accuracy. The computational speed of the bot is significant enough to have minimum delay in transmission of data. An application to alert the user is also developed in order to view the data from the robot. Therefore, this surveillance bot can be used to patrol indoor areas with maximum efficiency.

VI. Future Scope

Surveillance is a field with constant advancements, as the data gets more confidential and important, the methods to protect it also gets complicated and expensive. This particular project provides surveillance at a certain security level like face recognition to detect intrusion and exchange data between user and the bot. With more man power and expense, the bot can be developed in such a way that the bot can live stream the data directly to the user, who can view the stream using the application and hence take necessary actions if needed. The robot can also have a battery indicator in order to view the amount of voltage left in a battery before it can be replaced. Furthermore, the bot can be developed in a way that the bot is capable of taking decisions by its own, if there is an intrusion happening in its vicinity. With development in chassis, the bot can be deployable in multi terrain regions for outdoor surveillance.

VII. References

[1] Baras N, Nantzios G, Ziouzios D, Dasygenis M. Autonomous obstacle avoidance vehicle using LIDAR and an embedded system. In2019 8th International Conference on Modern Circuits and Systems Technologies (MOCAST) 2019 May 13 (pp. 1-4). IEEE.

[2] Setiawan E, Muttaqin A. Implementation of K-Nearest Neighbors face recognition on low-power processor. Telkomnika. 2015 Sep 1;13(3):949.

[3] Mutiawani V, Rahmany S, Abidin TF. Anti-theft vehicle monitoring and tracking android application using firebase as web service. In2018 International Conference on Electrical Engineering and Informatics (ICELTICs) 2018 Sep 19 (pp. 72-77). IEEE.

[4] Rivai M, Hutabarat D, Nafis ZM. 2D mapping using omni-directional mobile robot equipped with LiDAR. Telkomnika. 2020 Jun 1;18(3):1467-74.

[5] Dharmasena T, Abeygunawardhana P. Design and implementation of an autonomous indoor surveillance robot based on raspberry pi. In2019 International Conference on Advancements in Computing (ICAC) 2019 Dec 5 (pp. 244-248). IEEE.

[6] Abdelaziz MA, Salman AD. Mobile Robot Monitoring System based on IoT.

[7] Nayyar A, Puri V, Nguyen NG, Le DN. Smart surveillance robot for real-time monitoring and control system in environment and industrial applications. InInformation Systems Design and Intelligent Applications 2018 (pp. 229-243). Springer, Singapore.

[8] Shanavas H, Ahmed SA, Hussain MH. Design of an Autonomous Surveillance Robot Using Simultaneous Localization and Mapping. In2018 International Conference on Design Innovations for 3Cs Compute Communicate Control (ICDI3C) 2018 Apr 25 (pp. 64-68). IEEE.

[9] Bokade AU, Ratnaparkhe VR. Video surveillance robot control using smartphone and Raspberry pi. In2016 International Conference on Communication and Signal Processing (ICCSP) 2016 Apr 6 (pp. 2094-2097). IEEE.