International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 12 | Dec 2021 www.irjet.net p-ISSN: 2395-0072

Signal Jamming Autonomous Rover

Manjunath Bhaskar[1] Pooja Manjunatha[2]

[1,2]*Final year Students, Dept. of Information Science and Technology, Vemana Institute of Technology, Bangalore, Karnataka, India*

Abstract – *A **Signal DE-authentication automated rover**, which will be able to maneuver on its own alongside a radio silence bubble. There is no signal transmission i.e no message can be sent or received if the user is within the range of the rover.*

*Most of the commercial **UGV's** rely on spread spectrum techniques, such as direct sequencing and frequency hopping. We have implemented signal DE-authentication to make it efficient and affordable.*

*Our model is made to enhance its maneuvering accuracy by using **DQN algorithm**, **ultrasonic sensors** and **Node-MCU** for signal jamming.*

Keywords*: **Jammer, UGV, ROVER***

1. INTRODUCTION

In recent times, several attempts to infiltrate the military's network at the border have gone unnoticed and has been the cause of security and data breaches. However, even upon receiving alerts of possible breaches we have been able to classify the breach only by deploying a soldier to scan the vicinity because of which there have been innumerable casualties.

Evolving technology as such as has made it possible for us to tackle the situation by devising a locomotive that will not only minimize the casualties but also successfully gain ample amount of information about the target with accuracy and also remain stealth at the same time.

The Unmanned Ground Vehicle (UGV) is integrated with camera such as the Pi camera to be able to observe the vicinity during the day and night. The vehicle is intended to save the lives of soldiers as it performs critical and hazardous tasks currently handled by war fighters in the combat zone. The vehicle can maneuver using algorithm such as DQN with information that is obtained from the sensors.

2. LITERATURE SURVEY

[1] An Autonomous Robot for Intelligent Security Systems, Authors- Raza Hasan, S. Asif Hussain, Shaikh Azeemuddin Nizamuddin, Salman Mahmood

In the given system design, a robot system is built to monitor and identify the motion in achieving better security and surveillance of indoor environments. This autonomous system is built by using an embedded system to perform specific tasks and function as defined. The drawbacks here that we found was the bandwidth as it is shard with many UGV's another drawback is the lagging due to which it may occur to get faulty commands or it might even get mislead.

[2] Real-Time Object Detection with YOLO, Authors- Geethapriya. S, N. Duraimurugan, S.P. Chokkalingam

Object detection is a technology that detects the semantic objects of a class in digital images and videos. One of its real-time applications is self-driving cars. In this, our task is to detect multiple objects from an image

But only that won't be enough for detecting small objects for that we are using ultrasonic sensors too, in order to enhance the productivity of the detection of object.

3. DESIGN

The ROVER is designed to work under hazardous environment with military compliance.

Our effort here is to increase efficiency and reduce human error as sending a soldier when there is a security breach along the border might be harmful. During intimidating situations it becomes a liability to prevent causalities and information compromise. The intervention of machinery gives us leverage over such circumstances. UGV's also help in countering the threats from Explosive Ordnance (EO) that includes improvised explosive devices, and landmines thus preventing casualties. This device has repeatable precision at all times and can be much more accurate than humans; they may have micro inch accuracy. ROVER'S have sensors which can have capabilities beyond that of humans and can process multiple stimuli or tasks simultaneously, humans can only one. This

System replaces human workers for utmost security and surveillance. The power subsystem is a significant place as it estimates the working time and avoids the shutdown of power all of a sudden. This state requires a block for converting the AC-DC power for charging and backup of the system. The ROVER has motor drivers, two DC servo motors for locomotion.

3.1 WORKING

For object detection YOLO algorithm is implemented .This data is then sent for senor processing along with the data received by the Ultrasonic Sensors.

The proposed system has two main phases: Maneuvering and Signal jamming

The rover navigates from point A to point B with the help of deep learning algorithms such as YOLO (Convolutional Neural Networking algorithm) and DQN .The raspberry pi detects and classifies objects using Pi camera and calculates the distance with the aid of ultrasonic sensors. Direction control decisions are thus made by the rover with all the above information.

Upon detecting a network, the Node-MCU proceeds to DE-authenticate and jam the network. It also generates an open rogue access point to which the target network connects to.

4. IMPLEMENTATION

Object detection and movement

YOLO algorithm is used for real-time object detection. An **ultrasonic sensor** is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves and converts the reflected sound into an electrical signal. The Raspberry Pi detects and classifies objects using the **Pi Camera** on board and calculates the distance with the aid of ultrasonic sensors.

HC-SR04 distance sensor is commonly used with both microcontroller and microprocessor platforms like Arduino, ARM, PIC, Raspberry Pie etc.

L298N 2A is a high power motor driver perfect for driving DC and Stepper motors. The simulators used are CARLA and AWSDeepRacer.

Step 1: Load the CFG, weight and coco file

Step 2: Load the image

Step 3: Convert to RGB

Step 4: Resize the image

Step 5: Object detection

Step 6: Print the objects and the confidence levels

Step 7: Plot the image with bounding boxes

4.1 DQN Algorithm

Our aim will be to train a policy that tries to maximize the discounted, cumulative reward

$$R_{t0} = \Sigma\Sigma_{t=t0}\ \gamma\gamma_{t-t0}\ r_t$$

Where R_{t0} is known as the *return*.

The main idea behind Q-learning is that if we had a function Q^*: *State*×*Action*→R . We are constructing a policy that maximizes rewards $\pi^*(s) = \arg a \max_a Q^*(s,a)$

For our training update rule, we'll use a fact that every Q*Q* function for some policy obeys the Bellman equation:

$$Q^{\pi\pi}(s,a) = r + \gamma\gamma\ Q^{\pi\pi}(s', \pi\pi(s'))$$

The difference between the two sides of the equality is known as the temporal difference error.

$$\delta = Q(s,a) - (r + \gamma \max_a Q(s',a))$$

To minimize the error, we use Huber Loss:

4.2 Streaming

The live streaming is performed by the pi camera and pi network.

4.3 Collision avoidance

The output from the pi camera and the ultrasonic sensors are used to help the rover avoid collision while maneuvering.

The images are captured in the PI-camera.

Using these images, image classification technique is done by the YOLO algorithm.

The distance between the object and the rover is calculated by the ultrasonic sensors.

The rover is then self-taught to avoid collision on the field with the objects.

Step 1: Start

Step 2: Transmit signal at 37 kHz

Step 3: if echo is received:

Calculate object distance

If distance < 40:

Turn right

Else:

Continue on route; turn 90^0 right

REPEAT

Step 2

else: GOTO Step 1

Step 4: Stop

```
def create_model(self):
    base_model = Xception(weights=None, include_top=False, input_shape=(IM_HEIGHT, IM_WIDTH,3))
    x = base_model.output
    x = GlobalAveragePooling2D()(x)
predictions = Dense(3, activation="linear")(x)
    model = Model(inputs=base_model.input, outputs=predictions)
    model.compile(loss="mse", optimizer=Adam(lr=0.001), metrics=["accuracy"])
    Return model
```

4.4 Signal Jamming

Signal Jamming is the deliberate jamming, blocking or interference with wireless communications.

The concept can be used in wireless data network to disrupt information flow.

Jamming is performed by NodeMCU Wi-Fi Jammer - deauther using an ESP8266 board.

The NodeMCU detects a network and successfully de-authenticates the node from the network. The opponent will not be able to send any data to its base and hence the information is lost.

4.5 Report Generation

Upon Successfully detecting and Jamming a network that sees to pose a potential threat, the ROVER additionally generates an overall automated report of the suspected network activity and the beacons it captured while jamming the network successfully or not in the case of monitoring.

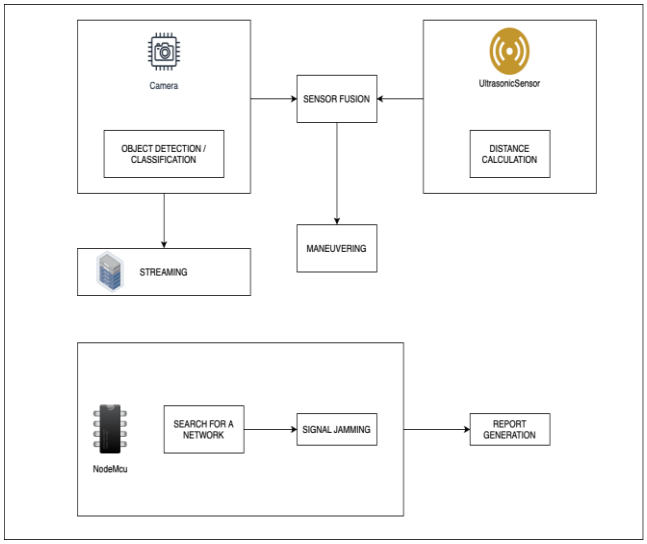


FIG 1: Overview of the system

5. SYSTEM REQUIREMENTS

5.1 Functional Requirements

The functional requirements of the system are:

- The rover should be able to maneuver with ease.
- The rover should be able to discover networks within its radius and, gain statistics and successfully jam the network.

5.2 Non-Functional Requirements

The functional requirements of the system are:

- Efficiency: The system is made to perform at the best possible efficiency.
- Uniformity: The system must perform uniformly on all inputs.
- Speed: The system is built to provide least possible latency while increasing the throughput and response time.

5.3 Software Requirements

The different software requirements required by the application are as follows:

- Operating System: Microsoft Windows 7 or later, Ubuntu 14.x or later
- Architecture: 32-bit or 64-bit versions are required
- Language: Python 3.6, C++.
- IDE: Spyder, Jupyter Notebook, Notepad++

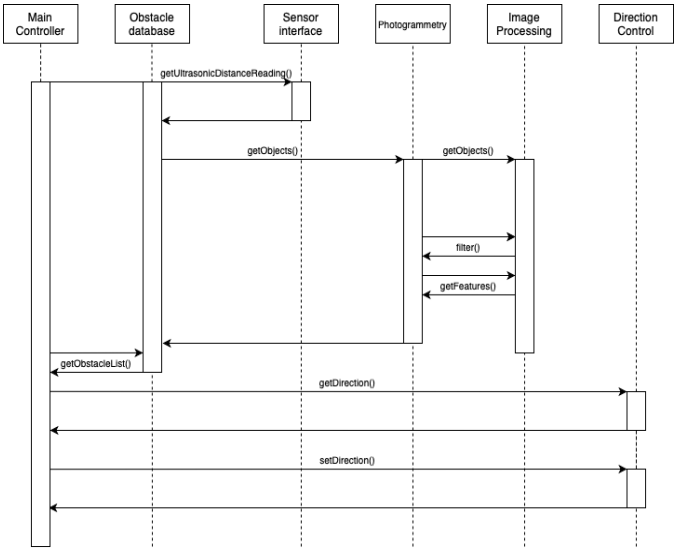
5.4 Hardware Requirements

The following describe the hardware requirements for ideal running of the application:

- Processors: 2.2 GHz Intel Core i3 and above
- Memory: 8 GB 1600 MHz DDR3
- Storage: 500 GB Hard-disk
- NodeMCU
- Ultrasonic sensor
- Pi camera
- Raspberry pi 4

The overview of the system is represented in Figure 1 above. It shows the different modules involved in building the system. Raw data is captured by the pi camera, ultrasound sensors and is processed. The required parameters from this processed data is filtered and is given as an input to the sensor fusion for further decision making. Based on the output of sensor fusion and current driver state, the rover accelerates or decelerates in that particular direction. This real-time application can be viewed on a display monitor.

The Raspberry pi initiates real time video recording and sends it to the pi camera, this module returns real time video data for detecting objects and classify them using their parameters. Ultrasonic sensor upon receiving the 'initiate_transmit_pulse' signal from the raspberry pi, a pulse is transmitted and is echoed for calculating distance by the pi. This data from pi camera and ultrasonic sensor is used for maneuvering. Network setup commands are given to Node-MCU. Upon detecting a network, the Node-MCU proceeds to DE-authenticate and jam the network by generating an open rogue access point to which the the target network connects to. A report on the jammed network is sent to raspberry pi. Detection of networks takes place simultaneously as the ROVER is in motion, upon detection, the ROVER stops and performs jamming.



The main controller gets the ultrasonic sensor readings from the sensor interface via the getUltrasonicDistanceReading() function and calculates the distance. Obstacle database fetches the objects from pi camera which is processed to detect and classify objects. The objects captured by pi camera are detected and classified and these objects are saved in the obstacle database. Raspberry pi uses the getObstacleList() function to get a list of objects that are detected and classified. The distance calculated earlier in correlation with the object list, tells us how far the objects are, then the main controller sets the direction in which the ROVER has to move.

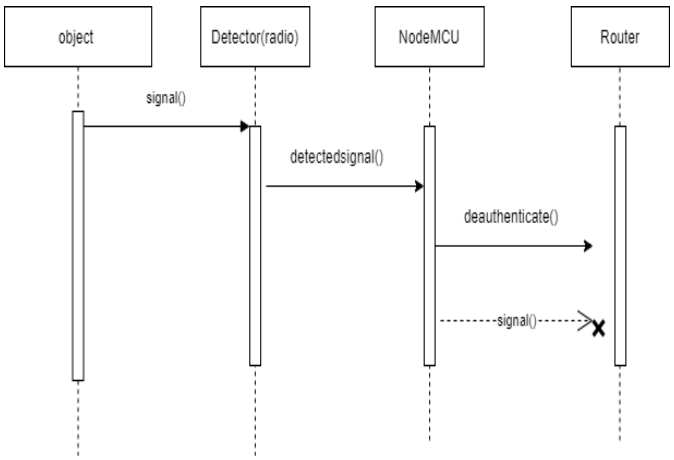


Fig 1.4: Network Setup

The detector (radio) detects the opponent's signal, if any and notifies the NodeMCU. The NodeMCU then deauthenticates the opponent's signal from the router it was supposed to reach.

Note - NodeMCU does not perform signal jamming. It DE-authenticates the client from its access point. Therefore, the message does not reach the opponent

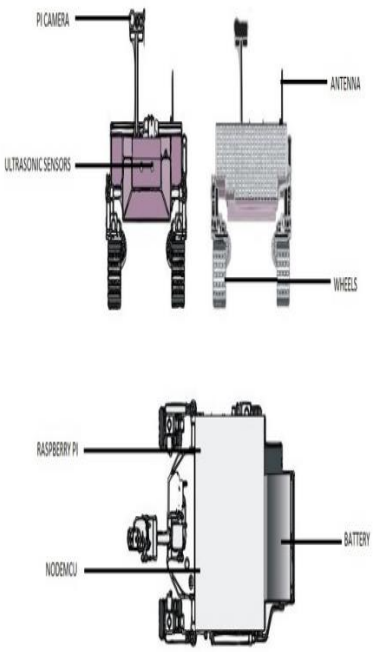


Figure 1.1 : Connection of parts

In view of the objectives of our project, we have been successful in implementing all aspects of our rover. Mobility wise, the rover is equipped with a driver that runs its motors and to enhance its decision making, it is aided by the Ultrasonic sensor along with visual inputs that help the rover is overcoming its shortcomings and making accurate decisions along the course of the path it traverses. The NodeMCU has been made use of to set up an interface to have an overview of all networks in the vicinity, narrow down on an opponent that seems suspicious and then proceed to carefully launch an attack on that user by DE-authenticating him from the network.

6. FUTURE SCOPE

Can use Signal jammers with the permissions from the government and enhance it. With proper training of the model the ROVER would achieve higher rates of accuracy

7. CONCLUSIONS

This project was initiated to overcome the inability of humans to detect harmful networks and successfully jam them. The UGV is intended to save the lives of soldiers as it performs critical and hazardous tasks currently handled by war fighters in the combat zone. The flexibility to install multiple payload systems increases efficiency and reduces life-cycle costs and soldier workload. Cell jammers, signal blockers, GPS jammers, text jammers, holds up the radio frequency in a given area, creating a sort of signal traffic jam that blocks all communication. Like a radio silence bubble, no calls or texts can be sent or received as long as the user is within the range of the cell phone signal blocker.

ACKNOWLEDGEMENT

We would like to express my special thanks of gratitude to our guide "Miss. Vijaya S.C" and "Miss. Stella" for their able guidance and support in completing our project.

REFERENCES

[1] Raza Hasan , Shaik Asif Hussain , Shaikh Azeemuddin Nizamuddin , Salman Mahmood "An Autonomous Robot for Intelligent Security Systems"- 2018 9th IEEE Control and System Graduate Research Colloquium (ICSGRC)

[2] Geethapriya. S , N. Duraimurugan , S.P. Chokkalingam "Real-Time Object Detection with YOLO"- International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8, Issue-3S, February 2019

[3] Keiron O'Shea, Ryan Nash "An Introduction to Convolutional Neural Networks" - Submitted on 26 Nov 2015 (v1), last revised 2 Dec 2015 (this version, v2)] - Cornell University

[4] Akshaya U Kulkarni , Amit M Potdar , Suresh Hegde , Vishwanath P Baligar "Radar detection using ultrasonic sensor" - 25-27 July 2019 1st International Conference on Advances in Information Technology (ICAIT)

[5] Pinal Rupani , Prof. Naren Tada " Literature Survey on Jamming Attack in Wireless Ad-hoc Network" - 2017 IJEDR | Volume 5, Issue 2 | ISSN: 2321-9939

BIOGRAPHY

Manjunath Bhaskar
BE Graduate ISE,
Vemana Institute of technology,
Bangalore-34

Pooja Manjunatha
BE Graduate ISE,
Vemana Institute of technology,
Bangalore-34