MOBILE REMOTE SURVEILLANCE TOWER

Yash Kale[1], Kirtan Chauhan[2], Harshad Jain[3], Siddhesh Chavan[4], Supriya Dicholkar[5]

[1-4]BE Student, Electronics and Tele-Communication, Atharva College of Engineering, Mumbai, India
[5]Professor, Electronics and Tele-Communication, Atharva College of Engineering, Mumbai, India

---***---

Abstract -*The project is the combination of the main software as well as hardware technologies. The tower is mobile which can be controlled with the Bluetooth so we can place the tower in the high terrain areas. The project is just a prototype of our idea but gives desirable outputs. The tensorflow model is used for the detection and classification of the objects from the frame. The object if detected is brought to the center creating an alarm, on the respective army base station. The objects classification is of wide 90 classes but for our purpose we have limited it to human and vehicle. This can be changed as per need. This all process is done by the raspberry pi 3B+, also we have used OpenCV for the displaying messages on the frames, capturing of images, and drawing detection boxes on the frame. The project gave us desirable outputs as per the hardware and software combinations*

Key Words: Object detection, OpenCV, TensorFlow

I.INTRODUCTION

The main aim of this project is to protect the border areas of the nation from the trespassers and the intruders in the nation. The illegal activities carried out near the borders can also be suspected through vigilance of the nearby area from a high terrain. This tower has a wide range of application spectrum. The vision comes to mind because despite having the second largest armed force in the world, we face intruders coming across the borders on the regular intervals and we lose the life of our brave soldiers.

II.GENERAL DESCRIPTION

II.A. BLOCK DIAGRAM

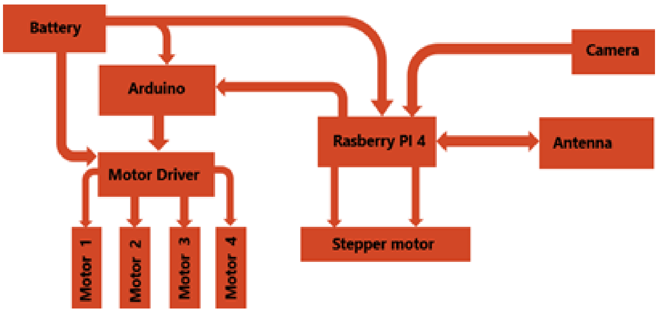


Figure 1

II.B. OBJECT DETECTION

The working of the system starts with setting up tensorflow's object detection with the Raspberrypi3 B+. After setting up this we would be able to use Raspberry Pi 3B+ to perform object detection on live video feeds from the camera on the tower.We have to train the neural network model which will then help us to identify specific objects, and then we would use raspberry-pi for unique detection applications. The model we'll use is the ssd_mobilenet_v1_coco_2018_01_28 from the TensorFlow detection zoo model. This model is Google's collection of pre-trained object detection models that have various levels of speed and accuracy and this model is trained on coco 2017 dataset. This model is trained on the ssd algorithm. The raspberry pi3B+ has a weak processor, so we need to use a model that takes less processing power. This model runs fast with high accuracy as per our observation. The object detection script detects objects in live feeds from a Pi-camera or USB webcam. The script first initializes the camera position, and then begins performing object detection on each video frame. You can get different color box frames on the human and vehicle and also count them in the frame with few changes in the code script. Raspberry-Pi 3B+ performs fairly well, achieving a frame rate nearly 1fps. The frame rate can be increased further on the basis of the camera and the Raspberry Pi processor advancement.

II.C. ROVER MOVEMENT

For the movement of the rover we need to define input and output pins for Raspberry-Pi. Next we need to initialize GPIO pins for Raspberry-Pi. Next, we need to define functions for the user to control the movement of the rover. Another function needs to be allotted which takes input from the user. Various Keys have been assigned to perform specific tasks. Key W is assigned for forward movement, key S is assigned for backward movement, key D is assigned for right movement, key A is assigned for left movement and finally key X is assigned to stop. Each key when pressed followed by pressing enter will perform the task associated with it and move in its desired direction.

II.D. HARDWARE IMPLEMENTATION

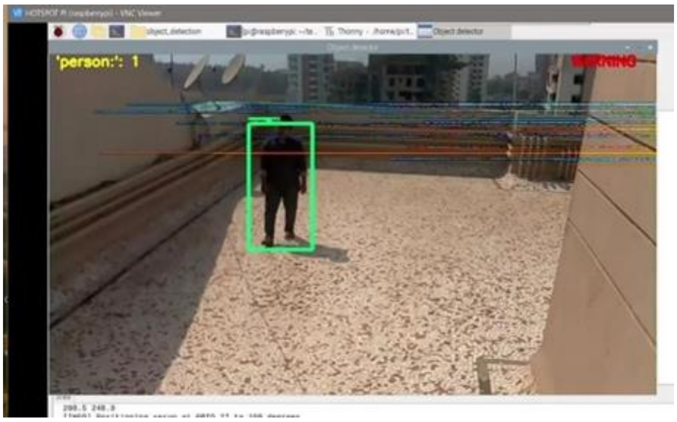
It is a mobile tower which can be elongated so it is easy to transport and take it to high terrain as well .It is automated, so as it reaches the spot it will take the shape of a tower. We have one stepper motor at the bottom of the rod which while rotating will bring it to a desired height we have set it for different height levels. While traveling, the rod can be shortened and can get inside the box so it will be easy to travel. The project is divided into two parts. First is the rover part (i.e. mobility) and then the transmission of data by video processing. The base of the tower will be controlled by Arduino, which is controlled by Raspberry Pi 4. The base has gear motors which are powered by motor driver and controlled by arduino so they can climb certain terrain having some weight on. The raspberry pi will take the input from the IP camera then it is sent to the remote center for further process. The camera will just record the video and transfer the data to raspberry pi4. It will stream the video to a remote location. The power of all this will be provided by 18650 LI-PO CELL.

II.E. SOFTWARE IMPLEMENTATION

The software implementation consists of the streaming of video on the remote area and the object detection on it. Initially we have to set up the Raspberry Pi for camera interaction. We have to install TensorFlow and then OpenCV. TensorFlow's object detection usually uses matplotlib to output images, but we will use OpenCV because it's handy and has less error chances. Then we go for protobuf and path directory setting. The TensorFlow object detection API uses Protobuf, a package that implements Google's Protocol Buffer data format. Then we set the paths and then we have to download the SSD_Lite model from the TensorFlow detection model zoo. Which is Google's collection of very good pre- trained object detection models that have different levels of speed and accuracy. The Raspberry Pi doesn't have a strong processor, so we will use a model that takes less processing power, compromising on accuracy. So we decided to use SSDLite-MobileNet, which we think is the fastest model available at the present moment. Then we will have certain rectangular boxes of different colors for the person detection and for the vehicle detection. When they are in the video frame of the camera they will be boxed giving an alarm to the person monitoring the region with the help of the tower.

III. RESULTS AND OUTPUT

The following project gave us the satisfying and desirable output as per the combined specifications of the raspberry pi, tensorflow model used. Since we have used raspberry pi 3B+ in our project we get lower fps of 0.7. The alarm sound and the display of the messages on the frame are smooth, having a quick response time. The movement and detection works absolutely fine. The detection is of high accuracy and is fast since the model is very well trained. The rover movement controlled with the Bluetooth is also smooth and has a good range of more than 20 meters. The rover is programmed and works fine to rotate on the same location to either left or right direction. We can make our project smoother by using Raspberry pi 4, Coral USB Accelerator and Tensorflow light model so we can get 34fps.

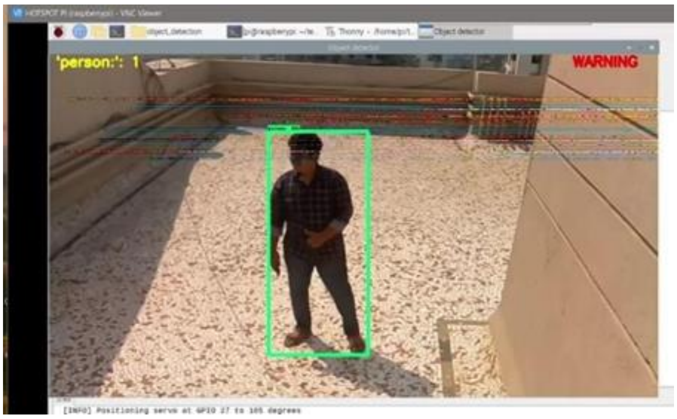


'person:': 1

WARNING

```
208.5 248.8
[INFO] Positioning servo at GPIO 27 to 100 degrees
```

Figure 2



'person:': 1

WARNING

```
[INFO] Positioning servo at GPIO 27 to 105 degrees
```

Figure 3

The output video as shown above gives us the output of the number of objects (vehicle/person).In each frame we have a detection box around it if the object box is detected .The counting is shown with class on the top left corner of the display image .The "WARNING" message is also displayed on the top right corner of each frame till the object is there in the frame. There is also an alarm sound played when the object is detected in the raspberry p; it stops when there is no detection. Below the screen there are readings printed about the pan and tilt servo angle's measures, with the center (x and y) of the rectangular detection box according to which the servos are moved to bring the object in the center of the frame.

IV. APPLICATIONS

For security and surveillance purposes along the border areas or even on private properties for trespassing.

V. CONCLUSIONS

We have drawn out some satisfactory results as a prototype of the idea. The results obtained are shown above in the table. There is some need for good hardware kits which will help us to accelerate the performance of the prototype .The processing speed is just one of the drawbacks but it can be rectified using coral usb accelerator .We have not used night vision camera but that will give you the similar kind of output in the specified range. This prototype can have many changes based on the conditions and it is adjustable according to that. The distance of the detection and accuracy can also be taken to greater level by having a camera of higher resolution.

Sr. N	Specifications	Output
1	Frames Per Second	0.7-0.8
2	Resolution	1280 X 720
3	Detection Distance	20m
4	Accuracy of Detection	Depends on the threshold value. We have kept 40%
5	Detection Response Time	< 0.5 seconds
6	Servo Range	X-axis:25 TO 140 degrees. Y-axis:20 to 130 degrees.
7	Bluetooth Distance	20m
8	Alarm Sound Response	< 0.5 seconds
9	Counting Accuracy	Excellent

Table 1

VI. FUTURE SCOPE

Many improvements can be made to make this more powerful and efficient like addition of TPU Coral Accelerator by Google, addition of gsm module to have a worldwide access from anywhere and it won't be just limited by the range of wifi. A powerful computer for more clear vision and faster response time. Night vision camera with a better resolution will enhance the usability further.

ACKNOWLEDGMENT

We would like to give our special thanks to our guide Prof. Supriya Dicholkar for mentoring us throughout our project. Also, we are thankful to the faculty of Electronics and Tele- Communication Department for assisting and helping us with our work whenever needed.

REFERENCES

[1]https://www.electronicsforu.com/electronicsprojects/make- video-streaming-camera-with raspberry-pi

[2] https://github.com/EdjeElectronics/TensorFlow Object-Detection-on-the-Raspberry-Pi

[3] Ajay Talele, Aseem Patil, Bhushan Barse, "Detection of Real Time Objects Using
TensorFlow and OpenCV" proceeding of Asian Journal of Convergence in Technology.'

[4] Onkar Kirpan, Pooja Bavisar, Shivani Khawase, Anjali Mankar, Karishma Ramteke, " Object Detection on Raspberry Pi" proceeding of IJESC

[5] Rajeshwar Kumar Dewangan, Yamini Chouhan, "A Review on Object Detection using Open CV Method"
proceeding of International Research Journal of Engineering and Technology (IRJET)

[6] Aneesha Maddineni, Dr.Sk.Jakeer Hussian, " Live Object Detection api using Tensorflow on Raspberrypi" proceeding of IJCRT

[7] R. Sujeetha, Vaibhav Mishra, "Object Detection and Tracking using Tensor Flow" proceeding of International Journal of Recent Technology and Engineering (IJRTE)

[8] Kuan-Hsun Wang, Shang-Hong Lai, "OBJECT DETECTION IN CURVED SPACE FOR 360-DEGREE CAMERA" proceeding of IEEE Conference on Computer Vision and Pattern Recognition

[9] Ivan Culjak, David Abram, Tomislav Pribanic, Hrvoje Dzapo, Mario Cifrek, "A brief introduction to OpenCV" proceeding of MIPRO 2012

[10] M. Naveenkumar, A. Vadivel, "OpenCV for Computer Vision Applications" Proceedings of National Conference on Big Data and Cloud Computing (NCBDC'15)