**Software Development Project**

**Staircase Detection**

By

**Motasim Billah Mredul**

**Roll**: 1507070

&

**Md. Ahsan Habib**

**Roll**: 1507082



**Supervisor:**

**Md. Milon Islam**

Lecturer

Department of Computer Science and Engineering

Khulna University of Engineering & Technology Signature

**Department of Computer Science and Engineering**

**Khulna University of Engineering & Technology**

**Khulna 9203, Bangladesh**

**Acknowledgment**

With the blessings and limitless mercy of Almighty, we are able to do this. We express our heartiest gratitude to Almighty Allah for this.

A word of special thanks must go to our highly esteemed teacher and this project’s supervisor, Md. Milon Islam, Lecturer, Department of Computer Science and Engineering, Khulna University of Engineering & Technology (KUET), for his excellent advices, guidance and right directions without which our project may not have reached a state it is in now.

We would like to thank our seniors who inspired us to implement different Ideas throughout this project. We would also like to thank our friends for their association also.

Any constructive comments, suggestions, criticism from teachers as well as seniors will be highly appreciated and gratefully acknowledged.

**Abstract**

The process of staircase detection is a complex process. Therefore, an intelligent system is required to detect staircase. In our project, we have developed a device using one ultrasonic sensor and one RGB-D camera to detect and recognize ﬂoor and both up and down staircases. The performance of an object recognition system depends on both object representation and classiﬁcation algorithms. This process requires high level of knowledge about image processing and working process of the algorithm.

In our system, we have used the ultrasonic sensor to measure the distance and detect staircase as well as the camera with a CPU where a model was trained in the TensorFlow library would detect if staircase is front of the user. Thus, creating ways for many applications in the future for both industrial purpose and helping visual impaired people and even in the dark to detect staircase.

**Contents Pages**

|  |  |
| --- | --- |
| 1. Introduction | 06 |
| 2.Objectives | 06 |
| 3. Previous Work | 07 |
| 4. System Requirements | 08-11 |
| 5. System Overview | 11 |
| 6. System Architecture | 12 |
| 7. Implementation | 12-15 |
| 8. Current Prototype | 16 |
| 9. Result Analysis | 17-18 |
| 10. Limitations | 19 |
| 12. Future Work | 19 |
| 13. Conclusion | 19 |
| 14. References | 20 |

**List of Figures**  **Pages**

|  |  |
| --- | --- |
| Figure 1. Raspberry Pi 3 Model B. | 08 |
| Figure 2. Ultrasonic Sensor Model HY-SRF 05. | 09 |
| Figure 3. Ultrasonic Sensor Model HY-SRF 05. | 10 |
| Figure 4. Webcam. | 10 |
| Figure 5. Block diagram of objection detection system. | 11 |
| Figure 6. Control flow of the system. | 12 |
| Figure 7. Workflow of TensorFlow. | 13 |
| Figure 8. Labelling image with upstairs. | 14 |
| Figure 9. Labelling image with downstairs. | 14 |
| Figure 10: (a) Upstairs detection (b) Downstairs detection using ultrasonic sensor. | 15 |
| Figure 11. Prototype of the system. | 16 |
| Figure 12. Upstairs detection using pre-trained model. | 17 |
| Figure 13. Downstairs detection using pre-trained model. | 18 |

**1. Introduction**

According to the World Health Organization (WHO), about 253 million people are visually impaired. Among them, around 36 million people are blind and rest 217 million people have various vision impairment. Among the above 80%, people are 50 years aged or above. As the world is progressing at a faster rate, new systems are being developed every day to make sure daily living are getting comfortable in every way possible. But for the people who have physical disabilities, they require more help than ordinary people. The domestic space and the urban environment contain various obstacle of different types at different locations. The detection of staircase offers valuable high-level knowledge to a variety of intelligent systems. Even for non-visually impaired people such congestion of such obstacle sometimes poses problems. So, for the visually impaired people those create more problems. Robot systems navigating in the public space can use such knowledge to expand their range of operation to multiple floors. For this to happen, these obstacles have to identified uniquely and then decide how move around these obstacles.

Our goal was to develop a real time device that can detect the staircase with a wide range of application. This project is a demonstration of a system with camera and ultrasonic sonar sensor used to detect staircase while merging these two sensing components to decide on the result.

**2. Objectives**

* Our target is to help both the visually impaired people and the robot technologies so that they could detect both upstairs and downstairs in multi floor building and also in the outside with steps like staircase.
* The main priority of the system is to detect staircase in the real time with both camera and ultrasonic sensor. The merging of the two sensing components was required as both the device had their limitations.
* We developed a system that could detect staircase in any kind of scenario.
* The system is a real time device and both the sensor can classify between up and down stair with high accuracy.
* The system can detect staircases in almost all possible cases. The previous built systems had only one features. The vision-based technologies could not work in the dark but our device can do that with ultrasonic sensor. Also, some devices were built using the sonar sensor, which deviates when there is too much noise around. But our device would detect staircase in both cases.
* A wi-fi connection is used to merge the components. So much accurate results can be computed by the output of the sensor.

**3. Previous Work**

In the last decade, many works have been done in this field to help the visually impaired people in different directions. Some of systems are in mobile devices, some for robot applications and the others for the visually impaired people using computer vision technology.

J. Razavi and T. Shinta [1] developed a new approach for staircase detection to aid the visually impaired and blind people. The system can detect upward and downward stairs distinctly. It used ultrasound sensor to determine whether it is upstairs or downstairs. The system is set to user’s head and used a reference distance that would correspond to flat ground. In terms of upstairs, the measured distance would be less than reference distance and in case of downstairs, the measured distance would be more than reference distance. A small fluctuate may occur because of user’s head moves up and down which is mitigated by using gyroscope. The main components used in the system are an ultrasound sensor, a gyroscope, an Arduino Nano, and two buzzers. The prototype of the proposed system is shown in Figure 2. The system is lighter and more efficient which weight is roughly 140g. This system has simpler interface, longer battery life which is around 18 days, low cost and low error rate and able to detect staircase even in dark, foggy, rainy and snowy conditions. However, the system may generate false alarms when the user with taller heights.

T. Schwarze and Z. Zhong [2] introduced a system for detection and tracking of upward-staircases to assist the visually impaired people and for the robots in multi-floor exploration scenarios. The system used libViso library for computing the six Degree of Freedom (DoF) motion of a moving mono/stereo camera. Random sample consensus (RANSAC) algorithm produces an initial plane by repeatedly sampling planes through 3 random points. The disparity images are acquired with OpenCV semi-global matching (SGBM) at half resolution. It also applied a canny edge detector to remove edge crossing and connect points extracted by flood fill region growing. The components used in the system is stereo camera. The proposed method is light-weight and works in a large range (up to 7-meter distance) to detect upward stairs. However, the system only detects upward staircase. It cannot detect downward staircases and pedestrians.

Pham et al. [3] developed a real time obstacle detection system in indoor environment for the visually impaired people that can help individuals to avoid four types of obstacles. The obstacles are walls, doors, staircases and a residual class that covers loose obstacles on the floor. The system used RGB-D camera to capture both RGB images and depth maps at a resolution of 640x480 pixels with 30 frame per second. The color and depth are combined to create a point cloud that commonly used to represent three-dimensional data and represented by x, y and z geometric coordinate of an underlying sample space. Then it used Voxel Grid Filter to down sample the point cloud and Random Sample Consensus (RANSAC) algorithm is used for detection in point cloud data. The main components used in the scheme are a personal computer (PC), an RGB-D camera, a Tongue Display Unit (TDU), a Kinect sensor. The system architecture of the proposed scheme is illustrated in Figure 3. The proposed scheme accurately detects and warns about nearby obstacles and notify the visually impaired people using either audio feedback or sensory substitution device or vibration. However, the system is not reliable when the intensity of light is heavy. The system is more costly and heavier to bear and only able to detect obstacles in a small range.

However, all these prototypes had major setbacks. In our system we have tried minimize the setbacks of the previous projects.

**4. System Requirements**

**4.1. Raspberry Pi 3 Model B**

The Raspberry Pi is a series of small [single-board computers](https://en.wikipedia.org/wiki/Single-board_computer) developed in the [United Kingdom](https://en.wikipedia.org/wiki/United_Kingdom) by the [Raspberry Pi Foundation](https://en.wikipedia.org/wiki/Raspberry_Pi_Foundation) to promote teaching of basic [computer science](https://en.wikipedia.org/wiki/Computer_science) in schools and in [developing countries](https://en.wikipedia.org/wiki/Developing_countries). The original model became far more popular than anticipated, selling outside its [target market](https://en.wikipedia.org/wiki/Target_market) for uses such as [robotics](https://en.wikipedia.org/wiki/Robotics). It does not include peripherals (such as [keyboards](https://en.wikipedia.org/wiki/Keyboard_(computing)) and [mice](https://en.wikipedia.org/wiki/Mouse_(computing))) and [cases](https://en.wikipedia.org/wiki/Computer_case). However, some accessories have been included in several official and unofficial bundles. Here is an illustration of following Figure 1.

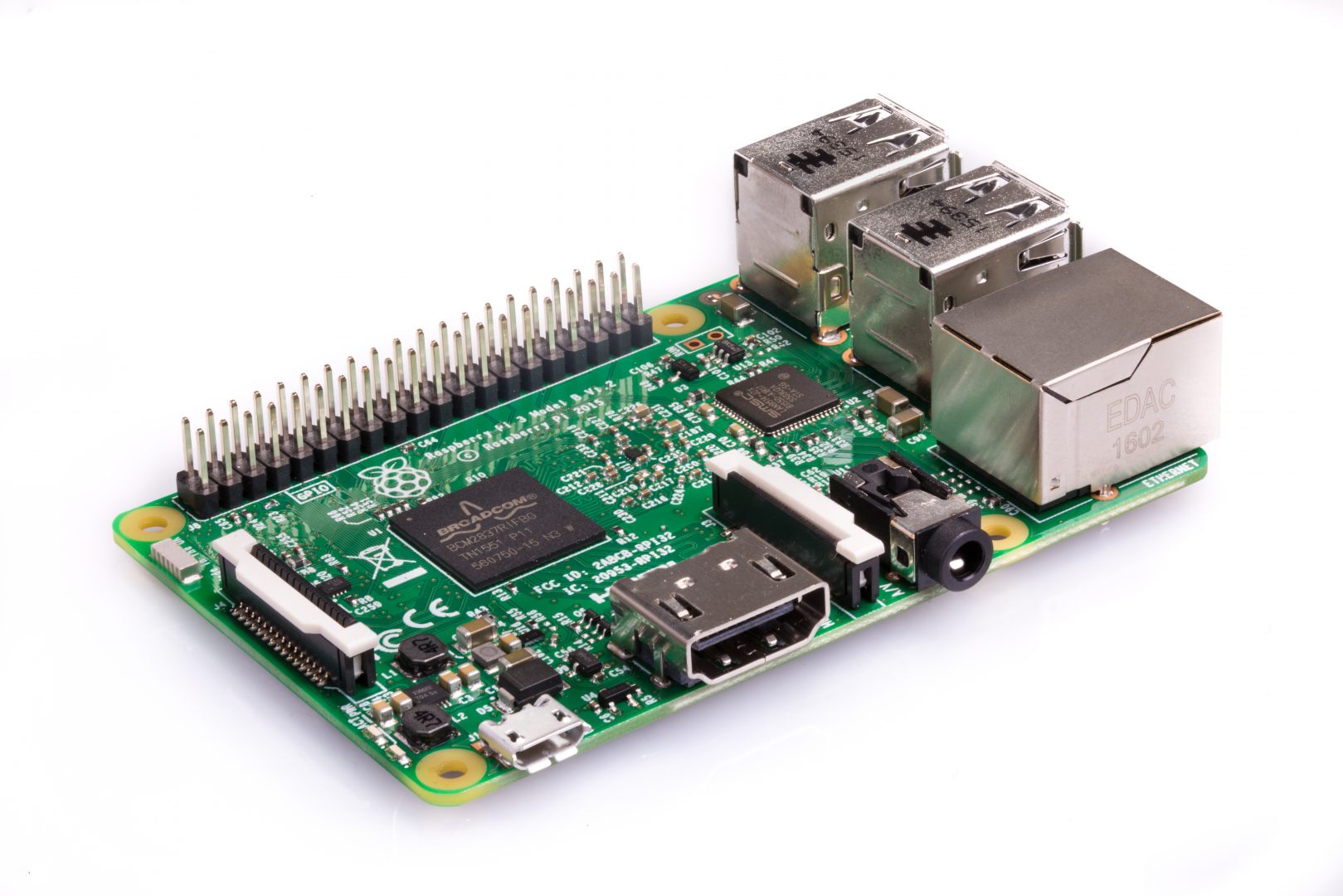


Figure 1. Raspberry Pi 3 Model B.

**4.2. Ultrasonic Sensor Model HY-SRF 05**

The SRF05 is an evolutionary step from the SRF04, and has been designed to increase flexibility, increase range, and to reduce costs still further. As such, the SRF05 is fully compatible with the SRF04. Range is increased from 3 meters to 4 meters. A new operating mode (tying the mode pin to ground) allows the SRF05 to use a single pin for both trigger and echo, thereby saving valuable pins on your controller. When the mode pin is left unconnected, the SRF05 operates with separate trigger and echo pins, like the SRF04. The SRF05 includes a small delay before the echo pulse to give slower controllers

The Ultrasonic module has two Transducers, one for Transmit & the other for Receive. Both are fixed on a single PCB with control circuit. The principle of ultrasonic distance measurement is the same as that of RADAR. From Arduino generate a short 10uS pulse to the Trigger input to start the ranging. The Ultrasonic Module will send out an 8-cycle burst of ultrasound at 40khz and raise its echo line high. It then listens for an echo, and as soon as it detects one it lowers the echo line again. The echo line is therefore a pulse whose width is proportional to the distance to the object. By timing the pulse, it is possible to calculate the range in inches/centimeters. If nothing is detected then the module will lower its echo line anyway after about 30mS. The module provides an echo pulse proportional to distance. If the width of the pulse is measured in μs, then dividing by 58 will give you the distance in cm, or dividing by 148 will give the distance in inches. μs/58=cm or μs/148=inches. The module can be triggered as fast as every 50ms, or 20 times each second. But there is needed 50ms wait before the next trigger, even if the SRF05 detects a close object and the echo pulse is shorter. This module is illustrated in followings Figure 2.

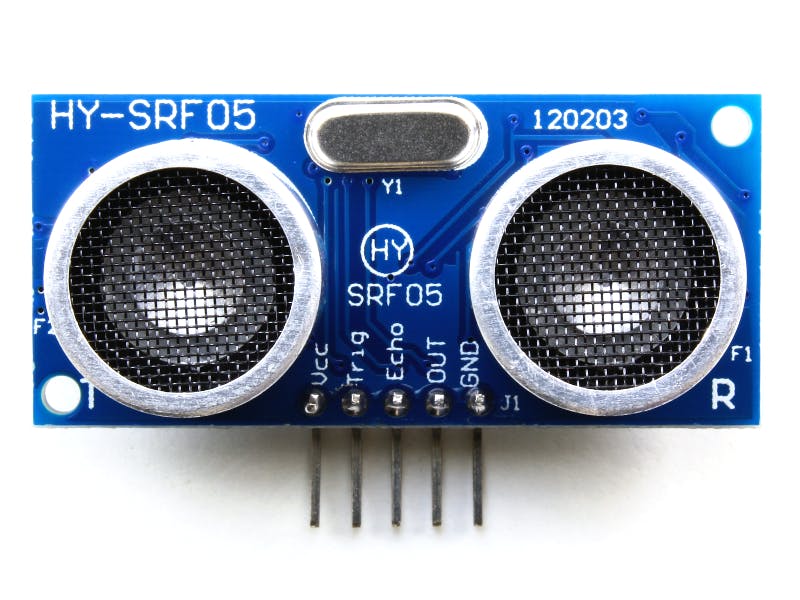


Figure 2. Ultrasonic Sensor Model HY-SRF 05.

**4.3. Buzzer**

Early devices were based on an electromechanical system identical to an electric bell without the metal gong. Similarly, a relay may be connected to interrupt its own actuating current, causing the contacts to buzz. Often these units were anchored to a wall or ceiling to use it as a sounding board. The word "buzzer" comes from the rasping noise that electromechanical buzzers made. Buzzers are depicted in following Figure 3.

|  |  |
| --- | --- |
|  |  |

Figure 3. Buzzer.

**4.4. Webcam (USB)**

A webcam is a [video camera](https://en.wikipedia.org/wiki/Video_camera) that feeds or [streams](https://en.wikipedia.org/wiki/Streaming_media) its image in real time to or through a [computer](https://en.wikipedia.org/wiki/Computer) to a [computer network](https://en.wikipedia.org/wiki/Computer_network). When "captured" by the computer, the video stream may be saved, viewed or sent on to other networks travelling through systems such as the internet, and e-mailed as an attachment. When sent to a remote location, the video stream may be saved, viewed or on sent there. Unlike an [IP camera](https://en.wikipedia.org/wiki/IP_camera) (which connects using [Ethernet](https://en.wikipedia.org/wiki/Ethernet) or [Wi-Fi](https://en.wikipedia.org/wiki/Wi-Fi)), a webcam is generally connected by a [USB](https://en.wikipedia.org/wiki/USB) cable, or similar cable, or built into computer hardware, such as laptops. Here is an illustration.



Figure 4. Webcam.

**5. System Overview**

The system has two parts with each part detecting staircase in their own different ways. The block diagram of the system is illustrated in following Figure 5.

Figure 5. Block diagram of objection detection system.

**Camera:** It is fixed on the top of the walking stick. The camera continuously captures images in front it and sends it to the trained model for detection of the staircase.

**Up/down stair detection through camera:** When the model was trained then there were two classes were created so that model could detect both the up and down stair. If no stair is detected in the given image then the model would mark the output as ground level.

**Ultrasonic sensor:** The ultrasonic sensor is attached to the walking stick in a fixed position with a reference was set by measuring the distance from ground from that fixed position. By comparing with that reference, the raspberry pi gives its output.

**Buzzer:** The buzzer was set on the walking stick to give signal to the user that the device has detected staircase ahead.

**6. System Architecture**

As it is mentioned before, our system has both came and ultrasonic sonar sensor, we combined the result from individual sensing components. Here the system architecture is depicted in following Figure 6.

**Continuous image**

**Buzzer**

**Ultrasonic Sensor**

**CPU**

**Wi-Fi**

**Raspberry Pi**

Figure 6. Control flow of the system.

At first images are taken using camera, then passes to CPU then predict the result and by the help of Wi-Fi the result passes to the raspberry pi where staircase is detected using ultrasonic sonar sensor. The raspberry pi merges the result and provide the output.

**7. Implementation**

The developed system has two main parts image processing consisting of an RGB-D camera and a CPU. The second part has ultrasonic sensor and a buzzer connected with raspberry pie.

**7.1. Camera Module**

The CPU has a pre-installed “TensorFlow” library in the python module. We used the Faster R-CNN (Region Convolutional Neural Network) inception v2 coco model which has a speed of 58 milliseconds and mean average precision of 28. We used more than 400 images for the training to get the attributes in the TensorFlow environment.

These images are pre-trained and provides results when the real time image is provided in the model. For this a high-speed processor was required for faster output. Here is the workflow of TensorFlow library.

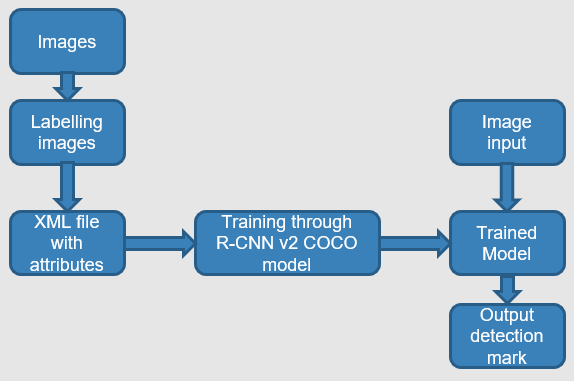


Figure 7. Workflow of TensorFlow.

We used “LabelImg”, a graphical image annotation tool, written in python and uses Qt for its graphical interface for labelling our training images. As our system can detect both upstairs and downstairs, we provided two categories of images for labelling. Here are some illustrations of labelling using the tool in Figure 8-9.

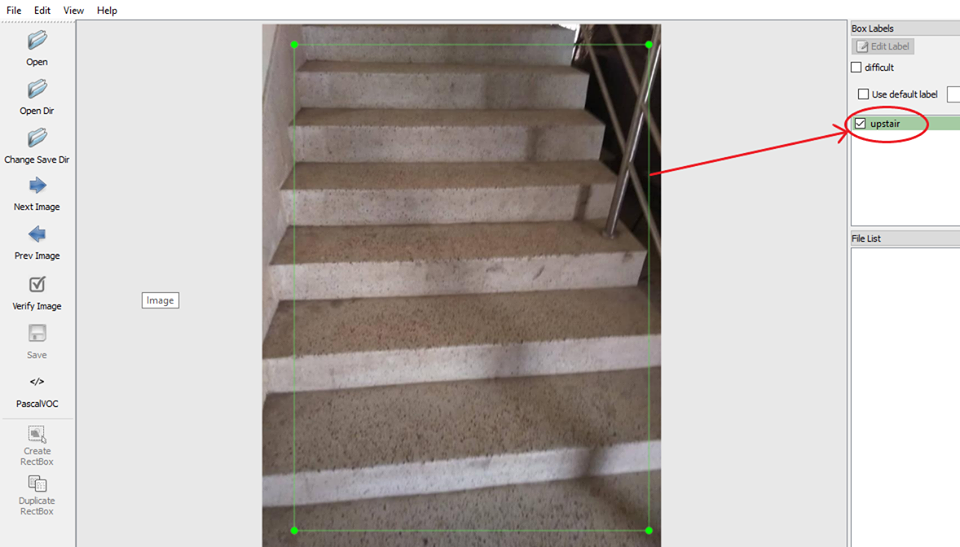


Figure 8. Labelling image with upstairs.

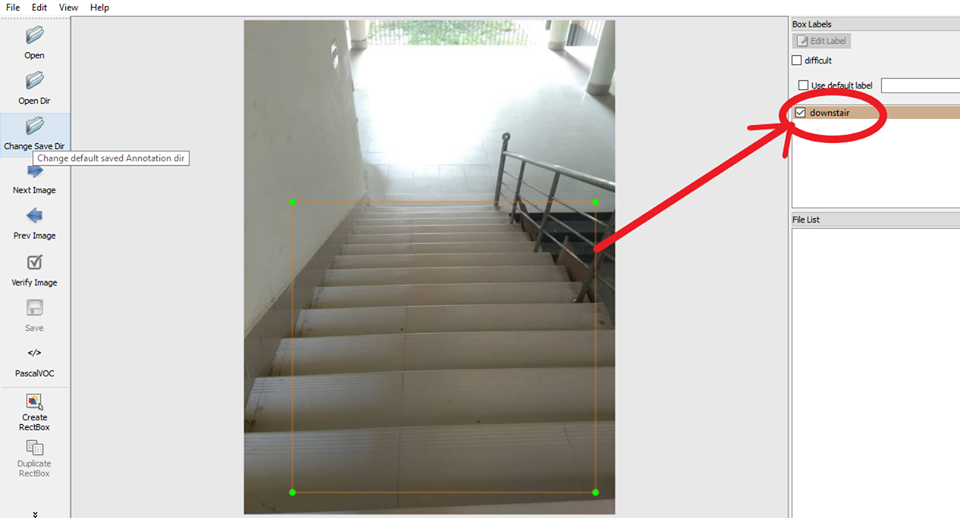


Figure 9. Labelling image with downstairs.

For upstairs images we used “upstair” class and for downstairs images “downstair” class. Annotation are saved as XML files in PASCAL VOC format.

**7.2. Ultrasonic Sensor Module**

The ultrasonic sensor senses the distance from its position to the obstacle ahead. Thus, giving the output of the distance of the obstacle which was used to predict possible staircase ahead.

|  |  |
| --- | --- |
|  |  |

Figure 10: (a) Upstairs detection (b) Downstairs detection using ultrasonic sensor.

When the power is given to the raspberry pi, the python script for the ultrasonic sensor starts execution and senses the distance continuously. Then the system establishes connection with the CPU through a Wi-Fi and waits for the result from there. We have set a reference for ground by measuring the distance of the ground from the sensor which is fixed. If the distance is greater the reference the sensor would detect downstairs and the opposite for upstairs. But the ultrasonic sensor deviates sometimes because of the voltage. That’s why we have taken 100 data each second and measured the mean to compare it with the reference.

When the camera is powered on it takes real time images from camera and each frame is send to the CPU for sensing if any staircase is ahead or not. The designed model can classify between the upstairs and downstairs. If any stair is detected, then it sends the information the raspberry pie.

If both the raspberry pie and CPU staircase ahead then buzzer is triggered on. There are two different sound for upstairs and downstairs.

**8. Current Prototype**

The current prototype of our system is depicted in following Figure 11.

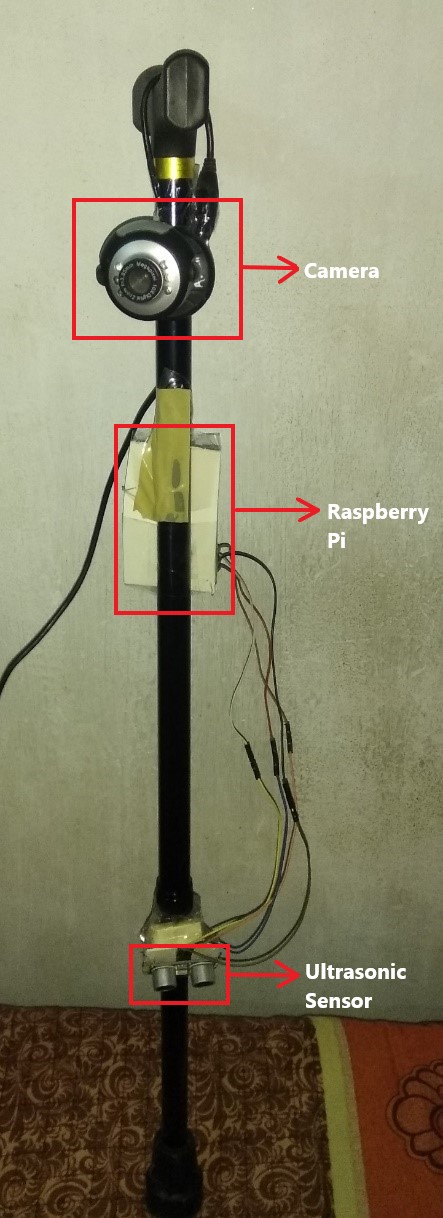


Figure 11. Prototype of the system.

**9. Results Analysis**

Our system can detect upstairs and downstairs using training model individually. Here are the examples of detection in following Figure 12-13.



Figure 12. Upstairs detection using pre-trained model.



Figure 13. Downstairs detection using pre-trained model.

On the other hand, using Ultrasonic sensor upstairs and downstairs can be detected individually.

As we have used two different sensors for detection of staircase both had different accuracy depending upon the circumstances. For the result of image processed from the model was giving as high as 99% accuracy in regular light condition. But when the amount of light decreases accuracy degrades to 60% to 50% both for the upstairs and downstairs.

For the ultrasonic sensors it depends on the amount of sound in the area of the user. When the sound is higher the result deviates. From our reading we have tested the prototype around 100 times where in result was found inaccurate around 20 % of times. But when the sound is higher the results was wrong around 60 % of times.

Here we used both camera and ultrasonic technology for better accuracy.

**10. Limitations**

* The camera cannot work the dark. Even in places where light intensity is very low, the accuracy in those circumstances degrades to 50-60% where the system accuracy was 98-99% if there is enough light.
* In the noisy places there is a high amount of deviation of data so result is not accurate enough sometime.

**11. Future Work**

* To make a firebase a model where the user will send the images in the server to compute the result in the server then show it in the device.
* To make the device power efficient.
* To count the number of steps in the staircase detected and the distance from the device.

**12. Conclusion**

Previously built devices on this field had some major issues in their model what we have tried to solve is those major concern about the cases in low light and noisy places. But the system is not fully perfect model as we have to carry the CPU with us for the computation of the images. Thus, making the device a bit complex for the user to carry. Also making it costly while thinking about all types of user. But the proposed device has better functionality than previous devices and is more user friendly than other devices built.

**References**

1. J. Razavi and T. Shinta, "A novel method of detecting stairs for the blind," in *Proc.* IEEE Conference on Wireless Sensors (ICWiSe), Miri, pp. 1-5, 2017.
2. T. Schwarze and Z. Zhong, "Stair detection and tracking from egocentric stereo vision," 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, 2015, pp. 2690-2694.
3. H.-H. Pham, T.-L. Le, and N. Vuillerme, “Real-time obstacle detection system in indoor environment for the visually impaired using microsoft kinect sensor, Journal of Sensors, vol. 2016, pp. 13, 2016.
4. LabelImg, <https://github.com/tzutalin/labelImg>.
5. Object Detection Classifier for Multiple Objects Using TensorFlow (GPU) on Windows 10, <https://github.com/EdjeElectronics/TensorFlow-Object-Detection-API-Tutorial-Train-Multiple-Objects-Windows-10>, Accessed on Nov 26, 2018.
6. Training Custom Object Detector, <https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/training.html>
7. Tensorflow detection model zoo, <https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md>, Accessed on Nov 2, 2018.
8. <https://www.tensorflow.org/>
9. How To Train an Object Detection Classifier Using Tensorflow 1.5 (GPU) on Windows 10, <https://www.youtube.com/watch?v=Rgpfk6eYxJA>, Accessed on Feb 19, 2018.
10. TensorFlow Models, <https://github.com/tensorflow/models>
11. How To Install TensorFlow (GPU v-1.4) on Windows 10, <https://www.youtube.com/watch?v=RplXYjxgZbw>, Accessed on Nov 5, 2017.
12. Intro – TensorFlow Object Detection API Tutorial (part1-part6) , <https://www.youtube.com/watch?v=COlbP62-B-U&list=PLQVvvaa0QuDcNK5GeCQnxYnSSaar2tpku>, Accessed on Aug 21, 2017.
13. A tutorial on implementing tensor flow object detection API with Webcam, <https://www.youtube.com/watch?v=MoMjIwGSFVQ&t=681s>, Accessed on Aug 25, 2017.
14. Ultrasonic Sensor SRF-05, <https://wiki.eprolabs.com/index.php?title=Ultrasonic_Sensor_SRF-05>, Accessed on May 23, 2016.
15. Buzzer, <https://en.wikipedia.org/wiki/Buzzer>, Accessed on Oct 24, 2018.