CSL 607: Multimedia Systems
Blind Assistant

Pratham Gupta
2015CSB1024@iitrpr.ac.in
Milan Chowdhari
2015CSB1010@iitrpr.ac.in

***Abstract*—People with complete blindness or low vision often have a difficult time self-navigating outside well-known environments. Traveling or simply walking down the road may pose great difficulty.**
The aim of our project is to create an application in order to help blind people in their day to day lives. We help to make their lives easier by performing some of the task in which they may face difficulties in life.

I. Introduction

The most important problems as perceived by the blind are difficulty reading bus numbers, inability to read street names, risk of road accidents. Statistics show that there are at least 70 legally blind involved in pedestrian-related accidents annually. They are also unable to access technology such as mobile phones normally.
We try to solve some of these problems via our application. We performs 3 major tasks which are:

1) Detection and tracking of incoming vehicles towards the person and warn him to prevent any accidents.
2) Detect and recognize any text present in an image taken by the person, and read it out to him.
3) Send the current location to friends or relatives in case of emergency via voice command.

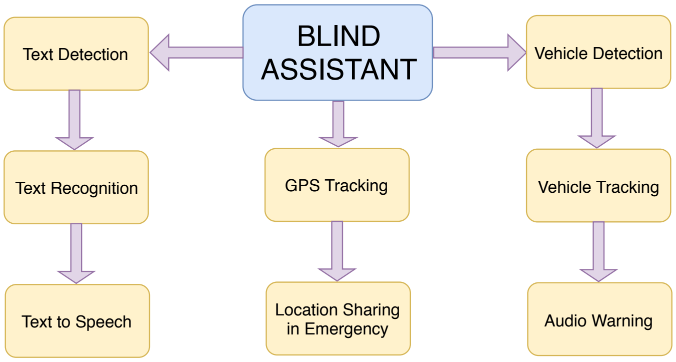

Fig. 1. Block Diagram

II. Vehicle Detection and Tracking

This part detects and tracks the vehicles in the view of the blind person and warns him beforehand if any vehicle is coming towards him, so that he can avoid an accident.

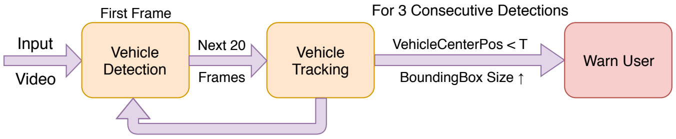

Fig. 2. Vehicle Detection and Tracking Methodology

A. Vehicle Detection

In this part we need to detect the vehicles and localize them using bounding boxes, in a frame of the video feed given by the user in real time. Since the system must work in realtime, this step needs to made as computationally fast and inexpensive as possible. For this purpose, we use a pre-trained YOLO (You Only Look Once) [1] model which has been trained on the ImageNet and the COCO datasets containing images from over a total of 9000 categories.
The model combines the region proposal network and the object classification network into a single deep neural network which allows for blazingly fast object localizations with a high enough accuracy. The model uses a sequence of blocks (convolution, leaky relu and maxpool layers) in succession to form a deep neural network. The output of the convolutional layers is passed into fully connected layers which can both detect and localize objects in one shot. So, all the vehicles in an image are detected and localized.
The bounding boxes obtained from the detection are passed on to the tracking algorithm to track the vehicles over the next frames.

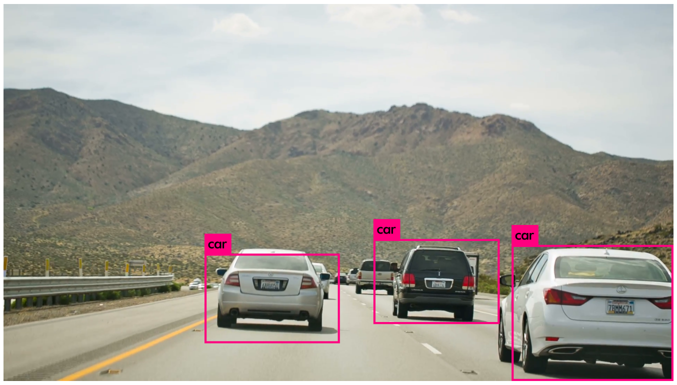

Fig. 3. Vehicle Detection

B. *Vehicle Tracking*

Now we track the vehicle in the subsequent frames using the detection in the first frame. Tracking is necessary as we need to do a real time analysis, and it is faster than detection algorithms as when we track an object detected in the previous frame, we know a lot about the appearance of the object. Since we need to do a real time analysis.
In this part we use the MEDIANFLOW Tracker [1] for the purpose of vehicle tracking. Internally, this tracker tracks the object in both forward and backward directions in time and measures the discrepancies between these two trajectories. Minimizing this ForwardBackward error enables them to reliably detect tracking failures and select reliable trajectories in video sequences. This tracker works best when the motion is predictable and small which is generally the case in vehicles as the motion between consecutive frames is minimal and the path of the vehicle can easily be predicted.

C. *Methodology*

The complete methodology of the tracking involves the following steps:

1) Detect all the vehicles in the first frame of the video feed.
2) Track the detected vehicles for the next 20 frames.
3) In each of the following frames check if the center of the vehicle changes position by more than a particular threshhold wrt the first frame. If not, the vehicle is coming towards the camera or moving away from it.
4) Ensure that the size of the bounding boxes is increasing in the subsequent frames to determine that the vehicle is coming towards the camera.
5) Vehicles detected again after 20 frames are matched to those in the previous section by thresh holding the intersection over union of the new and the previous vehicle bounding boxes. Then the same steps are repeated for the next 20 frames.
6) If a vehicle is coming in a straight line towards the camera in three consecutive sets of frames, we warn the person to move aside as the vehicle is coming towards him.
7) This method is repeated for each detected vehicle for all sets of frames.

III. Text Detection, Recognition and Speech

In this part we detect any text present in a particular image, recognize it and then convert the text to speech to read it out to the blind person. This can help him read road signs, showroom names in malls, shop names in a market etc.

A. *Text Detection*

Here we try to detect the text in any given image, and make bounding boxes around the detected text, to send it to the next part for recognition. The basic approach used to detect text blocks:

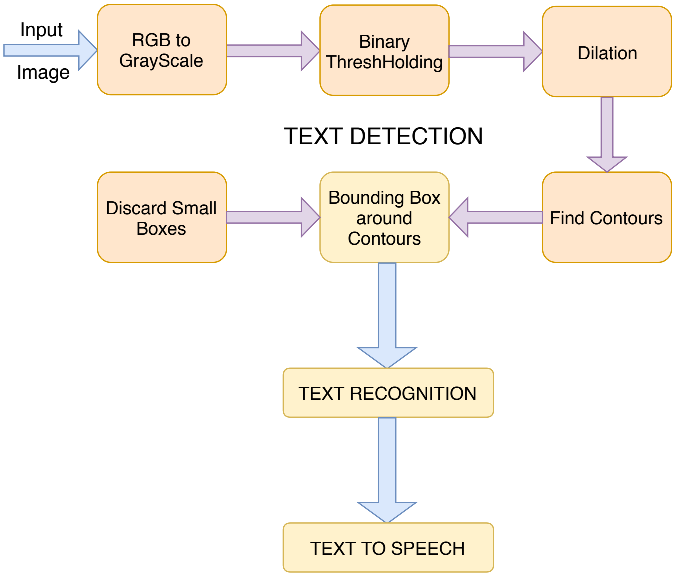

Fig. 4. Text Detection, Recognition and Speech Methodology

1) Convert the image to grayscale.
2) Apply a binary threshold, with a threshold value based on experiments.
3) Apply dilation to thicken lines in image, leading to more compact objects and less white space fragments.
4) Identify contours of objects in resulted image using opencv findContours function.
5) Draw a bounding box circumscribing each contoured object. Every block of text is framed in atleast one of the bounding box.
6) Discard areas that are unlikely to be text blocks given their size.

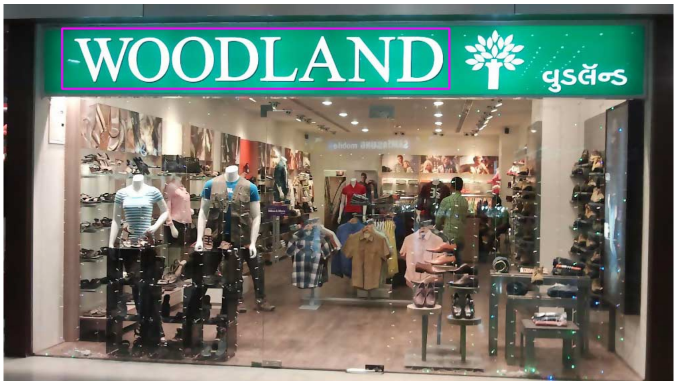

Fig. 5. Text Detection

B. *Text Recognition*

Now we recognize the text from the detected text blocks in the previous part. For this section we use Python-tesseract[2] an optical character recognition (OCR) tool for python. Optical character recognition (OCR) is the mechanical or electronic conversion of images of typed, handwritten or printed text into

[1] https://www.learnopencv.com/tag/medianflow/

[2] https://test.pypi.org/project/pytesseract/

machine-encoded text, from a scanned document, a photo of a document, a scene-photo etc. It works best in case of images with only text, which hence makes text detection necessary.

C. *Text to Speech*

We use the Google Text to Speech API [3] commonly known as the gTTS API for converting the recognized text to speech. It converts the text entered, into audio which is saved as an mp3 file. The gTTS API supports several languages including English, Hindi, Tamil, French, German and many more. The audio is then played to the blind person.

IV. GPS Tracking & Location Sharing

There are many cases when a blind person may get lost or find himself in an emergency. In such cases it is necessary for him to be able to inform about his location to his friends or relatives, so that they can come and help him. We try to integrate this feature in our application in a simple way so that he can send his location to his friends easily in case of an emergency. We use the Google Maps Android API in order to get the location data of the mobile. The person can simply click a button on the app and his location will be shared with the contacts decided beforehand via an email. The recipients can then help him, in whatever way possible.

V. Observation & Conclusion

We have observed that the vehicle tracking although fast, is not as accurate as detection, and the vehicle detection, although very accurate, is very slow on a CPU. Hence in order for the algorithm to work both accurately and fast, we need to reduce the number of frames for tracking and increase the frequency of detection. Also there is a need to implement the detection on a GPU as it can run much faster in than a CPU, and hence allow real time vehicle detection.

We can conclude that our app although not perfect but can be a very good help for blind people and make their lives easier by solving some major problems.

VI. Future Work

This project works as an all rounded assistant for blind people, and we would like to add some more features to help them in the future. One of the features would be image description, which will describe a scene to the blind person via audio, helping them better understand what is going on in their surroundings. We would also like to add a feature to detect other obstacles and help the blind people navigate their way on the road using a variety of sensors, by telling them the direction to move in, in order to avoid any obstacles. This will act as a replacement to the sticks they generally need to navigate their way.

References

[1] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018.

[3] https://pypi.org/project/gTTS/