Real time object detection for VI - (Project Blindly)

# Intro:

One of the most valuable senses for humans is vision. But, many of them were suffering with visual impairment and struggling to live a normal life along with others. They face difficulties in navigating independently and accessing information.

The idea of this project is to notify the blind people in audio form about the objects in their path. This will help them to navigate independently without others help by using real time object detection. This application uses image processing and machine learning techniques to determine the real time object by using a mobile camera and inform the blind people about the obstacles and its location through audio.

In addition to that the users are also notified of which directions they should move in, in order to avoid any obstructions in their path.

# App workflow:

1. Open the app on a mobile phone.
2. On launching the app, the camera will be launched automatically.
3. Then the camera will capture a real time view.
4. The image will be processed using OpenCV.
5. At each instant, the detected objects will be boxed with their confidence score.
6. The detected object and their location will be notified to the blind people.

# Object detection algorithm:

For efficient implementation, selection of algorithms is the most crucial part. Thus, different object detection algorithms like R-CNN, Fast R-CNN and YOLO are compared.

| Algorithm | Speed |
| --- | --- |
| R-CNN | 0.05FPS (20s/img) |
| Fast R-CNN | 0.5FPS (2s/img) |
| YOLO | 45FPS (22ms/img) |

# YOLO algorithm:

YOLOv3 is faster and more accurate compared to others. YOLOv3 dataset has 80 classes and 80000 objects. In one image more than 80 different objects can be detected by YOLOv3.

High performance is required for an image to be processed. Therefore, a dataset with a minimum number of objects is needed to run YOLOv3 on any mobile phone device without compromising on the accuracy of detection.

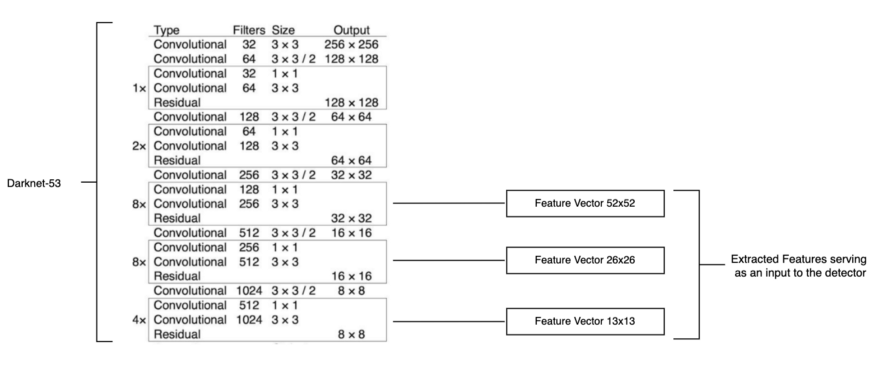
Predicting boundary box:

* Centre of a bounding box (bx ,by)
* Width (bw)
* Height (bh)
* Class of an object ( c )
* Probability of object (pc)

Y = (pc,c,bh,bw,bx,by)

For the detection of objects, YOLOv3 uses multi-label classification. Instead of using the softmax function, it uses independent logistic classifiers and threshold values.

For feature extraction, YOLOv3 uses Darknet-53 (a network trained on the ImageNet), and Residual networks(ResNet). The network uses 53 convolution layers (hence the name Darknet-53) where the network is built with consecutive 3x3 and 1x1 convolution layers followed by a skip connection.



# Drawbacks:

1. The object should not be far away from the camera.
2. The object should not be small.
3. The dataset should not be large to improve speed.
4. The detection of sun will be difficult.