Real-Time Mobile Application for Assisting Blind People Using YoloV2 and Google cloud vision API

Chitra Nambirajan Thevar
MCA Final Year
P. G. Department of Computer Science,
SNDT Women's University, Mumbai, India.

Abstract: Visually impaired people can't move safely outdoors because they cannot perceive the outside obstacles as normal people. The prototype application in this study aims to make the visually impaired people's lives easier with the mobile devices. The mobile application with the designed to see object nearby and read any text documents.The application is developed for the Android platform. Image processing and machine learning technologies are used.

Blind assistance is promoting a widely challenge in computer vision such as navigation and path finding. In this paper, two modules are designed text recognition and object detection are employed to provide the necessary information about the surrounding environment. Objects detection is used to find objects in the real world from an image of the world such as faces, bicycles, chairs, doors, or tables that are common in the scenes of a blind. Object detection is used to detect any obstacle at a medium to long distance. YOLOV2 machine learning algorithm is used to perform the object recognition and Google Vision Cloud API is used to perform Text Recognition. The proposed method for the blind aims at expanding possibilities to people with vision loss to achieve their full potential. The experimental results reveal the performance of the proposed work in about real time system.

Keywords **- Image Processing, Machine learning, Visually Impaired.**

I. INTRODUCTION

This study aims to produce an application prototype to make the life of visually impaired people a little bit easier by using intelligent mobile devices.

This system describes an Android-based application for object recognition and Text Reading developed to help the blind understand their environment better. This application is based on extracting local features of the object of interest, which are then matched to the corresponding features of objects saved in a knowledge base previously created. The local features are tested against more than one classification method and the results are analyzed for object detection. For Text reading in realtime camera frames are sent to Google Cloud Vision API for Text Recognition in English,Hindi and regional languages and text is converted to audio using gTTs (Google Text To Speech) whichreads text in real time.

II. OBJECT DETECTION WITH YOLOV2

- **OpenCV**

OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision. In simple language it is library used for Image Processing. It is mainly used to do all the operation related to Images.

1. Read and Write Images.
2. Detection of faces and its features.
3. Detection of shapes like Circle, rectangle etc. in a image. E.g. Detection of coin in images.
4. Text recognition in images. E.g. Reading Number Plates/
5. Modifying image quality and colors e.g. Instagram, Cam Scanner.
6. Developing Augmented reality apps.

"Computer Vision is a field of deep learning that enables machines to see, identify and process images like humans."

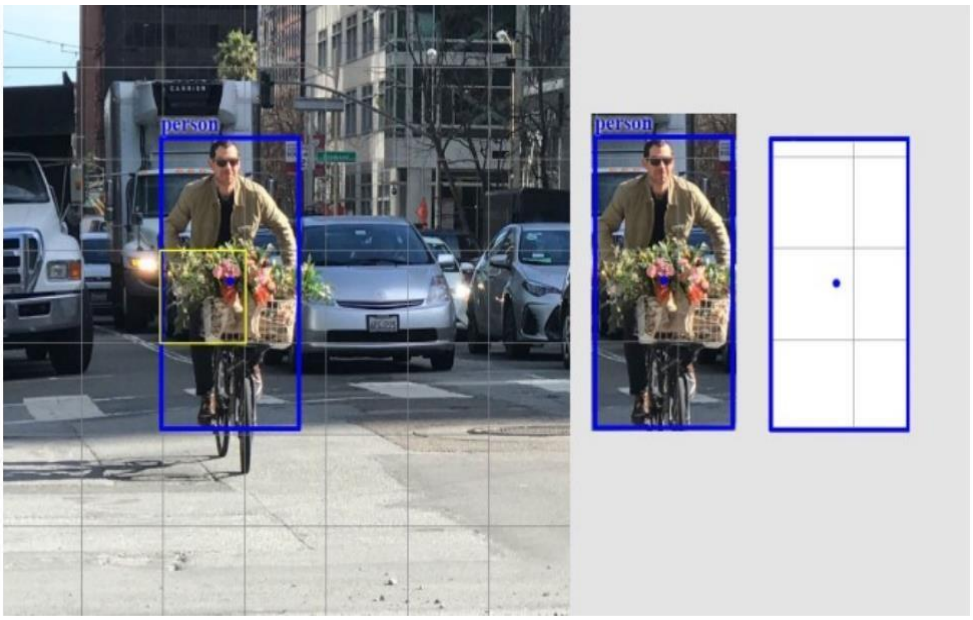


Figure 1 : Working logic of YOLOv2

Each grid cell of an image has a fixed number of boundary boxes. In Figure 2 the yellow grid cell generates two boundary box estimates (blue box) to find where the person is. Each grid cell detects only one object. In this study, high box confidence scores (greater than 0.25) is kept as our final predictions

Each boundary box contains 5 items: (x, y, w, h) and a box confidence score. The confidence score reflects an object of the box (objectness) and possibly how accurate the bounding box is. In operation, the bounding box width w and height h are normalized by the image width and height. x and y offset to the corresponding cell. Therefore x, y, w and h are all between 0 and 1. Each cell has 20 contingent class possibilities. The probability of a conditional class is the probability that the perceived object belongs to a particular class (one probability per category for each cell). So, YOLO's prediction has a shape of (S, S, B×5 + C) = (7, 7, 2×5 + 20) = (7, 7, 30)

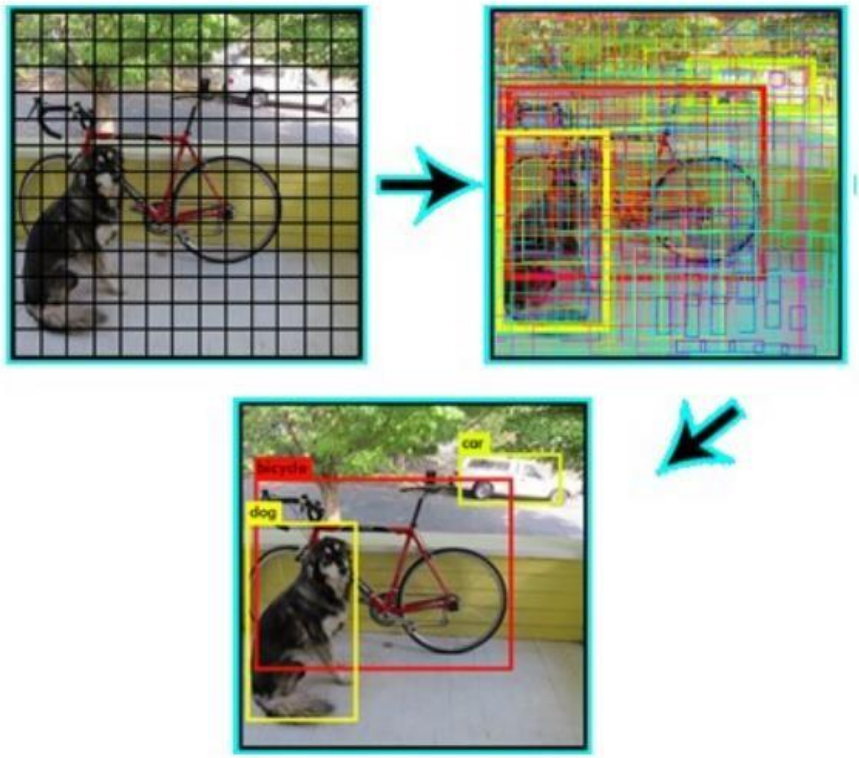


Figure 2 :Class confidence score = box confidence score x conditional class probability

YOLOv2 is faster and more accurate than prior detection methods. YOLOv2 dataset had 9,000 objects, it worked with low FPS even on the GPU processor. Image processing to be a process that requires high performance. So in order to run the dataset with the mobile phone processor, we had to work with a dataset with fewer objects. So in the study used the Tiny-YOLO dataset the next stage.

There is no account creation and no sign-in process. It is necessary to adjust the positioning of the mobile device so that it can see the full course direction in order to be able to carry out the identification well. In this implementation, GPS location data is not used. It performs the operation using the camera and audio output modules of the mobile device. It will recognize and display the defined objects in the application interface. There is also a certain perception distance to recognize objects. Objects recognized at the specified distance will first be reported to the user on the mobile device screen by showing a rectangular marker. The name of the object recognized on this rectangle will be displayed to the user.
Since the time between this object recognition and the notification of the user as a voice command is a millisecond event, the user will be informed directly without any delay. The intended use of the unit as a sound unit is to reduce the external noise, so that the user can hear the sound more clearly. And the user will be notified will be reported via the voice unit of the mobile device.

III. TEXT READING

Google Cloud Vision API was used for image analysis. Their project detects individual objects and faces within images, also finds and reads printed words contained within images. Paper evaluates the robustness of Google Cloud Vision API to input noise. In particular, Set of images are taken and noise is added to them then API is unable to detect correct text or object were as if the noise is removed then the output is similar to that of the original image. Cloud vision API can benefit from noise filtering

The paper proposed a prototype that helps people to hear the text content of the image in their native language. The text is extracted from the image and then text is converted to translate speech of user's native language. Camera captures the image and then OCR engine convert image to text. Then text is converted into speech using speak TTS engine. The speech output is then stored in a flac file. This file is then converted into desired language by the Microsoft Translator using a python script.

Paper proposed a system that reads text on a captured Image. It is performed as text is extracted in real-time from camera preview using Google Cloud Vision API and then converting the text to speech by gTTS tool.

IV. DATA AND SOURCES OF DATA

Dataset In this project is COCO Dataset. COCO 2014, containing 80 classes with approximately 40,000 training images. COCO is a large-scale object detection, segmentation, and captioning dataset.
This version contains images, bounding boxes " and labels for the 2014 version.
Note:

- Some images from the train and validation sets don't have annotations.
- Coco 2014 and 2017 uses the same images, but different train/val/test splits
- The test split don't have any annotations (only images).
- Coco defines 91 classes but the data only uses 80 classes.
- Panotptic annotations defines defines 200 classes but only uses 133.

YOLOv2 is a model trained to recognize specific objects. YOLOv2 with COCO dataset is used so it is faster and accurate than other algorithms.

Model	Layers	FLOPS (B)	FPS	mAP	Dataset
YOLOv1	26	not reported	45	63.4	VOC
YOLOv1-Tiny	9	not reported	155	52.7	VOC
YOLOv2	32	62.94	40	48.1	COCO
YOLOv2-Tiny	16	5.41	244	23.7	COCO

Table 1: Performance of each version of YOLO.

You Only Look Once (YOLO) was developed to create a one step process involving detection and classification. Bounding box and class predictions are made after one evaluation of the input image.
The fastest architecture of YOLO is able to achieve 45 FPS and a smaller version, Tiny-YOLO, achieves up to 244 FPS (Tiny YOLOv2) on a computer with a GPU.
The values of map and FPS of YOLOv2 and other datasets are given

Detection Models	Train	mAP	FPS
Fast R-CNN [2]	2007+2012	70.0	0.5
Faster R-CNN VGG-16 [3]	2007+2012	73.2	7
Faster R-CNN ResNet [4]	2007+2012	76.4	5
YOLO [5]	2007+2012	63.4	45
SSD300	2007+2012	74.3	46
SSD500	2007+2012	76.8	19
YOLOv2 288 × 288 [12]	2007+2012	69.0	91
YOLOv2 352 × 352 [12]	2007+2012	73.7	81
YOLOv2 416 × 416 [12]	2007+2012	76.8	67
YOLOv2 480 × 480 [12]	2007+2012	77.8	59
YOLOv2 544 × 544 [12]	2007+2012	78.6	40

Table 2. Detection frameworks on PASCAL VOC 2007

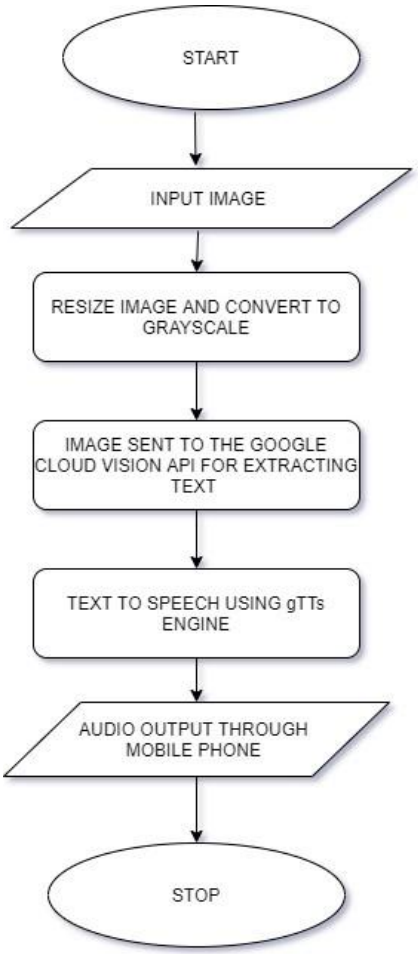
V. WORKING METHODOLOGY AND IMPLEMENTATION

A computer vision system basically consists of a camera to capture the image and a computing device which analyses the image captured.
The proposed system consists of a smartphone camera which works as an image sensor to capture the image. And lastly the Google Cloud Vision API is used which processes images on the Google Cloud Platform.

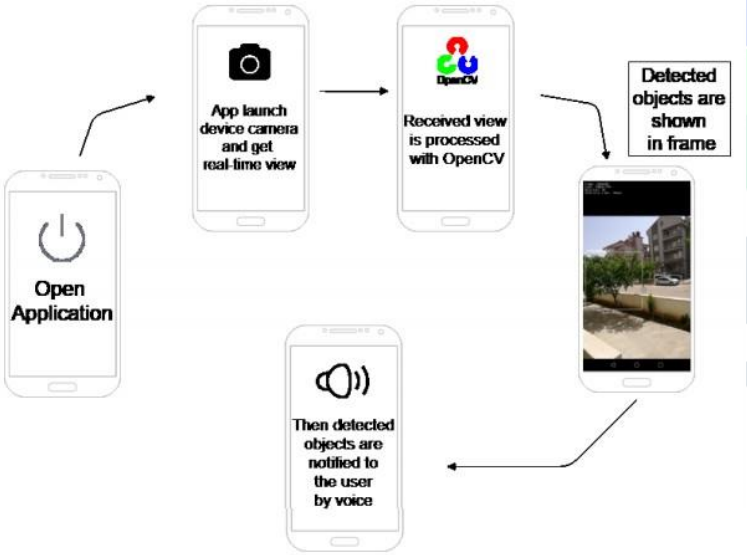
The other major component of the proposed system is the Google Cloud Vision API. It was released in the year 2015. The Google Vision API provides a RESTful interface that quickly analyses image content. This interface hides the complexity of continuously evolving machine learning models and image processing algorithms. It enables developers to analyze the content of images.
It uses powerful machine learning tools to extract the necessary data from images. It can perform different functions such as label detection, face detection, Logo detection, Optical Character Recognition (OCR) etc. Google Cloud Vision API's performance in the presence of noise is not very dependable and hence before sending the image to the Google Cloud.

The android application developed has a user friendly interface for visually impaired people. It consists of speech recognition activity by which the camera is started and the image is then resized to 720 x 480 pixels since greater the size greater is the time taken for OCR, after resizing the image is converted into grayscale. All the preprocessing steps were performed to reduce the time required for image to text translation.
The Google Cloud then performs OCR (Optical Character Recognition) on the processed (resized and grayscale) image sent to it. In the Google Cloud everything is encapsulated in a RESTful API which returns a bounding box (information about the location of text in an image with the help of x and y co-ordinates).

The Google Cloud can recognize different languages. The image resolution is so chosen that we get the correct output in minimum time. As the resolution of the image is increased the amount of time taken by the Google Cloud to perform OCR on it increases. The response obtained is in the form of JSON structure. The text is obtained and then converted into speech using the gTTS engine. And when user tap the screen it reads the text appeared on screen.



Flowchart : Text Recognition



Flowchart : Object detection

The steps of the implementation for object detection can be summarized as follows:

1. The objects that are needed to be defined at the application are determined.

2. Camera of the smartphone is used and the objects in the data set were identified on Android Studio. Profiles were created according to the data set and the voice commands for the detected objects are defined using gTTS..

3.Once Camera started user get to listen detected objects as well as objects are remarked in mobile screen using rectangle border.

VI. Results & Discussion

The dataset (Tiny YOLO) which is used in the tests, has worked very efficiently on Android. As you can see the Table 2, Tiny YOLO has also slightly lowered the mAP, but a much better FPS value is obtained. This value is sufficient for detection on the mobile device.
Tiny YOLO dataset values is shown in Table

Detection Models	Train	mAP	FPS
Tiny YOLO	COCO trainval	23.7	244

Table 3 : Performance on the Tiny YOLO dataset.

Currently there are totally 80 classes in the dataset. In the study, it is intended to further increase the classes in this dataset at the next stage.

Text is extracted from image and converted to audio. It recognizes different fonts. Skewed text images are also identified and converted into speech. The model recognizes the text which is readable by human eyes. System makes the use of a smartphone camera for better resolution pictures
The text-to-speech can change the text in image into speech with high performance of the images in which text is readable by the naked eye accurately.

As the resolution of image increases the time required for the processing also increases. Hence the images are first resized to 740 x 480 pixels and then the image is converted to grayscale in order to minimize the time required for image to text translation. If the preprocessing steps are skipped, then the time required increases to around 50 seconds and with the same result as above

VII. Conclusion

This study is designed to make visually impaired people more comfortable and aware of their daily life without any help from anyone. The visually impaired people will be able to notice the threats that may arise during transportation with voice feedback and this will help preventing possible accidents. The mobile devices can be carried easily and the camera of the device can be used as a third eye to the visually impaired people.

References

[1] OpenCV android https://opencv.org/android/

[2] Hossein Hosseini, Baicen Xiao and Radha Poovendran "Google's Cloud Vision API Is Not Robust To Noise" 16th IEEE International Conference on Machine Learning and Applications December 18-21, 2017.

[3] Davide Mulfari, Antonio Celesti, Maria Fazio, Massimo Villari and Antonio Puliafito "Using Google Cloud Vision in Assistive Technology Scenarios" IEEE Workshop on ICT solutions for eHealth 2016.

[4] Yolo RealTime Object detection – joseph Redmon. https://pjreddie.com/darknet/yolo/

[5] Google Cloud Vision API Android **.** https://cloud.google.com/vision/docs/samples

[6] Hui J., "Real-time Object Detection with YOLO, YOLOv2 and now YOLOv3", 2018, [Online]. Available: https://medium.com/@jonathan_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088
[7] YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers- https://arxiv.org/pdf/1811.05588.pdf
[8] Microsoft COCO: Common Objects in Context - https://arxiv.org/pdf/1405.0312.pdf

[9] "Vocal Vision Android Application for Visually Impaired Person"- International Journal of Science, Engineering and Technology Research (IJSETR) Volume 5, Issue 6, June 2016

[10] "Real Time Object Detection for Visually Challenged Persons "- International Journal of Innovative Technology and Exploring Engineering (IJITEE)