UNIVERSITY OF COLORADO DENVER
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FINAL PROJECT REPORT
TITLE: AI MODEL FOR BLIND PEOPLE

TEAM MEMBERS:
SANMATHI GURUPRASAD
NERELLA TARUN REDDY
PRATHIGA THIYAGARAJAN

ABSTRACT

In this project, we have tried to help people with visual impairment by providing a mechanism for object detection. Everyone knows just how difficult it is to be blind. Globally, at least 37 million people are estimated to have a vision disability or blindness according to the World Health Organization. Objects in the blind person's view are detected and their names are interpreted from the scene and translated into speech. The object's spatial positions are encoded into audio as output assistance. Video is taken with a portable Camera device and sent for real-time image recognition with existing object detection models. This camera-based device also assists blind people in reading text patterns written on items. It aids visually challenged persons in interpreting text patterns of text by translating it to audio output.

CONTENTS

TOPICS

1: INTRODUCTION

1.1 Smart Reading System for Visually Impaired People Using Tesseract

1.2 Flow Process of Tesseract

1.3 What is YOLO?

2: LITERATURE SURVEY

2.1 Literature Review

3: METHODOLOGY

3.1 Project Process

4: RESULTS

4.1 Object detection

4.2 OCR

5: CONCLUSION

BIBLIOGRAPHY

INTRODUCTION

Vision is God's best gift to people yet many people lag behind such ability. According to the 2019 WHO report, the global estimate of the number of visually disabled people is 37 million worldwide. This project aims to turn the visual world into the audio world with the potential for informing blind people about the objects and their spatial positions by converting them into speech. Video is captured with a client-side portable camera device and is streamed to the server with the existing object detection model YOLO (You Only Live Once) for real-time image recognition. The location of the items is determined from the location and the dimension of the bounding boxes. Our project also provides a smart tool that easily and effectively assists visually disabled people by reading paper-printed text. The camera must be put over the written material, the text will be read and the loudspeaker will give out the information.

1.1 SMART READING SYSTEM FOR VISUALLY IMPAIRED PEOPLE USING TESSERACT

We're presenting a smart device that helps visually impaired read the paper-printed text. A camera is used as a device that can be used for reading text documents. Based on studies with Blind people, the design is made portable. The proposed system feeds data into the system using a portable IP webcam which is then processed by Tesseract. The OCR software and the Text-to-Speech (TTS) are the fundamental blocks used as the basis for most access technology solutions designed for people with blindness and reduced vision.

Optical character recognition (OCR) -It is the translation into the machine-encoded text of the recorded images from printed text. OCR is a mechanism in which objects (letters, symbols, and numbers) relate a symbolic value to a character's image. Optical character recognition is also beneficial for people who are unable to read a text document but intend to know its content. OCR enables the use of machine translation, text to speech, and data extraction techniques in the documents recorded or scanned. The final accepted text document is fed to the output system or a speaker that can read out the text aloud.

1.2 FLOW PROCESS OF TESSERACT

1. Image Capturing

It is the step where the cell phone captures the image with text on it. To provide quick and consistent identification through the high-resolution camera, image quality should be high.

2 Pre-processing

Noise is removed in the pre-processing stage. The Image is scanned for skewing. Skewing can occur either in right or left orientation. The image is brightened before moving to further processing.

3 Segmentation

The picture is then transferred to the segmentation stage after pre-processing. In this process it attempts to break down a picture of a series of symbols into the individual symbol sub-image. The picture histogram helps to measure the horizontal line width. Width of the words is detected using histograms. They are then broken down into symbols that use symbol width calculation.

4 Feature Extraction

Feature extraction is the process of mining out features from a picture that are defined by character, height, and width, the horizontal and vertical lines, pixels in the various regions, etc.

5 Image to Text Converter

The ASCII values are processed for known characters. Here each character matches its equivalent pattern and is saved as a regularized transcript of the text.

1.3 WHAT IS YOLO?

The algorithm "You Only Look Once" uses convolution neural networks (CNN) to detect objects. YOLO is the quickest algorithm for object detection out there. It's a viable solution that is real-time and with little loss in precision. Similar to the recognition algorithms detection algorithms predict both class labels and object positions. So, not only does it classify the image into a classification, it can identify several objects within a picture as well. The Algorithm applies the full picture to a single neural network. It splits the picture into different areas and predicts boxes and probabilities for each area.

YOLO v3 uses a version of Darknet, trained on Imagenet with a 53 layer network.

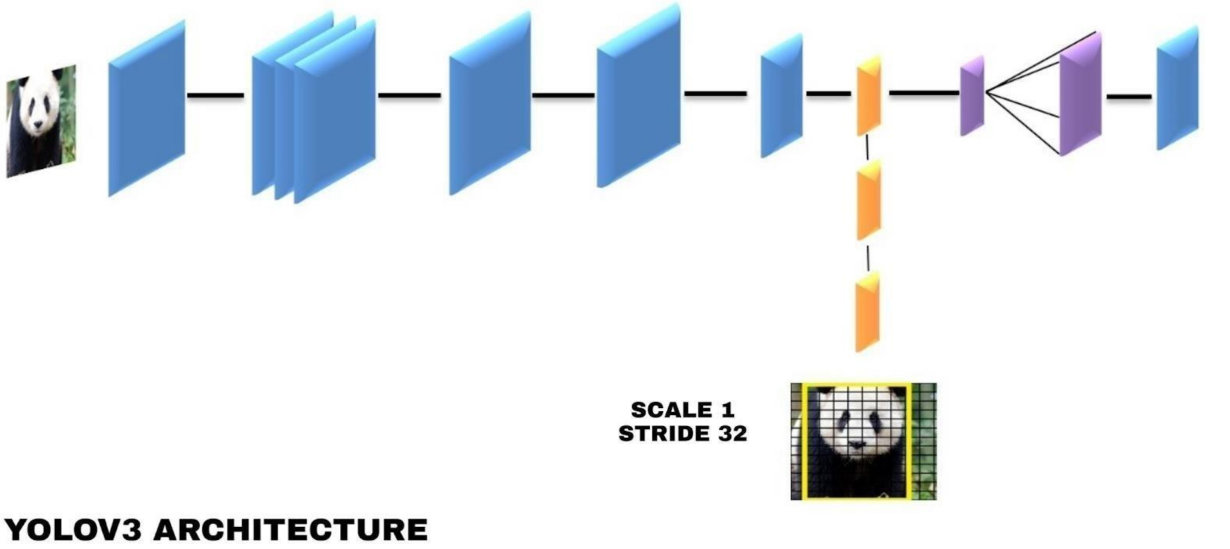


Fig 1.3 YOLOv3 Architecture

2 LITERATURE SURVEY

Kedar Potdar, Chinmay D.Pai, Sukrut Akolkar., 2018. [1] A Convolutional Neural Network-based Live Object Recognition System as Blind Aid. It uses a technique for the identification of live objects. It helps people with visual impairments who rely heavily on other senses such as touch and audio indicators to familiarize the environment around them.

Ferdousi Rahman, et al, 2018. [2] Assisting the Visually Impaired People Using Image Processing. The proposed thesis suggests detecting the light intensity and major colors in a real-time picture employing an external camera RGB method and thus identifying basic objects and facial recognition from individual data set. YOLO Algorithm and MTCNN Networking are used respectively for object detection and facial recognition. Support for the applications is accomplished by using Python's Free CV libraries.

Ajeet Ram Pathaka, Manjusha Pandeya, Siddharth Rautaray., 2018. [3] Application of Deep Learning for Object Detection. This paper clarifies the part of the deep learning techniques for object detection centered on a convolutional neural network. Also enunciated are deep learning frameworks and the services available for object detection. This paper assesses the profound learning strategies for object detection systems.

Ren, Shaoqing, et al., 2016. [4] Object detection networks on convolutional feature maps. IEEE transactions on pattern analysis and machine intelligence 39.7 (1476-1481). This paper shows that it's equally important to carefully design deep networks for object classification. They experiment with networks of regional classifiers that utilizes common, location-independent features and named them "Networks on Convolutional Feature Maps."

Redmon, Joseph, et al., 2016. [5] You only look once: Unified, real-time object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. They present YOLO, a modern Object Detection solution. Previous work on object detection stress on detection classifiers. Instead, they employ a regression technique. In one evaluation, a single neural network predicts bounding boxes and class probabilities straight from full pictures.

Mrs. Poonam Khare., 2016. [6] Literature survey on the various methods of object detection in video surveillance systems‖ International Research Journal of Engineering and Technology (IRJET) Here various phases of VSS are studied and alternate results with their benefits and drawbacks were discussed for each phase.

Simonyan, Karen, and Andrew Zisserman., 2014. [7] Very deep convolutional networks for large-scale image recognition. In this work they investigated the result of depth of the convolutional network on its precision in the setting of large-scale image identification. The key influence here is an exhaustive valuation of increasing depth networks by architecture with very small (3x3) convolution filters.

Epelea Laviniu, Gavrilu Ioan, Tiponu Virgil, et al., 2014. [8] OCR Application on Smartphone for Visually Impaired People. This paper explains the possibility of interaction between visually impaired people and a smartphone especially with the help of an android-based OCR application to replace the sense of sight. On the smartphone screen the program will accept voice commands or touch commands and the result is transmitted on speaker or headphones. The OCR process happens to be faster on a server on the internet.

Mishra, Nitin, et al., 2012. [9] Shirorekha chopping integrated tesseract OCR engine for enhanced Hindi language recognition. This paper presents a complete methodology for improving the accuracy of Hindi Language Recognition. This paper also demonstrates a comparison with other available Devanagari OCR engines based on the accuracy of identification, time of processing, font dissimilarities, and magnitude of data.

Unnikrishnan, Ranjith, and Ray Smith., 2009. [10] Combined script and page orientation estimation using the tesseract OCR engine. This paper demonstrates an algorithm for estimating the transcript's main page alignment in an image.

Ray Smith., 2007. [11] An Overview of the Tesseract OCR Engine. This paper explains The Tesseract OCR engine, as explained in an inclusive summary is the HP Research Prototype in the UNLV Fourth Annual Test of OCR Accuracy. Stress is made on features that are different or are uncommon in an OCR engine.

2.1 SUMMARY OF LITERATURE SURVEY:

Exploring the literature we have summarized that Object detection can be done majorly and efficiently using YOLO which uses a Neural Network for recognition of pre-trained objects on the COCO dataset. Besides, OCR uses well- trained models to identify each character. Tesseract is open source and has an OCR- based LSTM engine and adds several models for additional languages and scripts, making it to 116 languages in total.

3 METHODOLOGY

3.1 PROJECT PROCESS

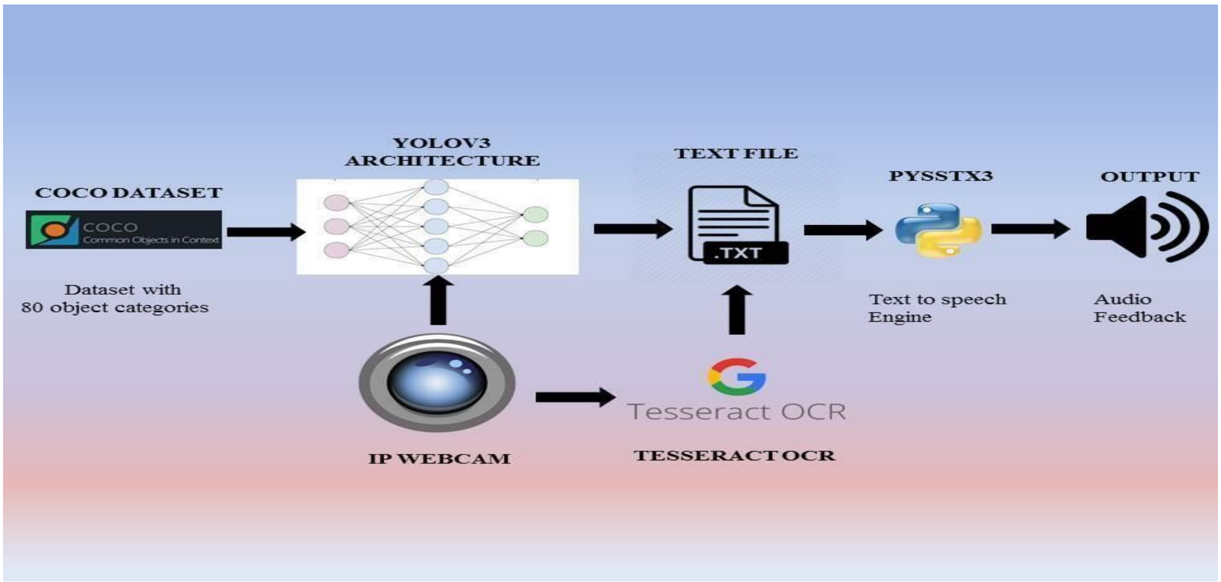


Fig 3.1: Process diagram

1. **Input**: We are utilizing IP webcam to feed pictures at 60 frames-per-second to this prepared model and we can also process alternate frames to add speediness
2. **Dataset**: The model is prepared with the Common Objects In Context (COCO) dataset.
3. **Model**: You Only Look Once (YOLO) which undergoes various perplexing Convolutional Neural Networks
4. **Text-to-Speech**: The class classification of the detected objects in each frame will be a string e.g. a cat. We will also obtain the object coordinates in the picture, and add the ―top/―mid/―bottom & ―left/―center/―right location to the ―cat class prediction. We use pysstx3 to handle the string to verbal output**.**
5. **Tesseract:** It is an OCR engine that is the best preferred and has a high graded OCR library. OCR makes use of AI for text finding and its recognition of the image. Tesseract- OCR is detecting templates in pixels, letters, words, and sentences.
6. **Output:** We get the bounding box coordinates for each object identified in our frames, and return the frame stream as a video replay. Voice feedback is scheduled on every 30th frame (30 fps) e.g. ―bottom left cat ― meaning a cat was detected in the camera view at the bottom-left.

4. RESULTS AND DISCUSSION

The framework proposed consists of two modules: object detection, and OCR. The key objective of object detection is to evaluate the presence of objects in the scene in front of users, while the OCR reads text to users.

The system performance was substantially very well for the gross positioning task. Objects within a few meters from the subject were effectively detected. Real-time processing while a continuous change in positioning was sometimes complicated but generally acceptable because every 30th frame is being captured and processed. It is reported that the verbal output for the detected objects was clear and intuitive to understand and is effective as it does not lead to information overload, and it takes less time getting used to. Then subsequently the ease of use was improved. Also during OCR detection, the text was recognized by the system very efficiently and which was then converted into verbal output with very high accuracy which also includes regional languages.

4.1OBJECT DETECTION:

The core module of this project is Object detection using YOLO. The following figure demonstrates the work-flow of this system.

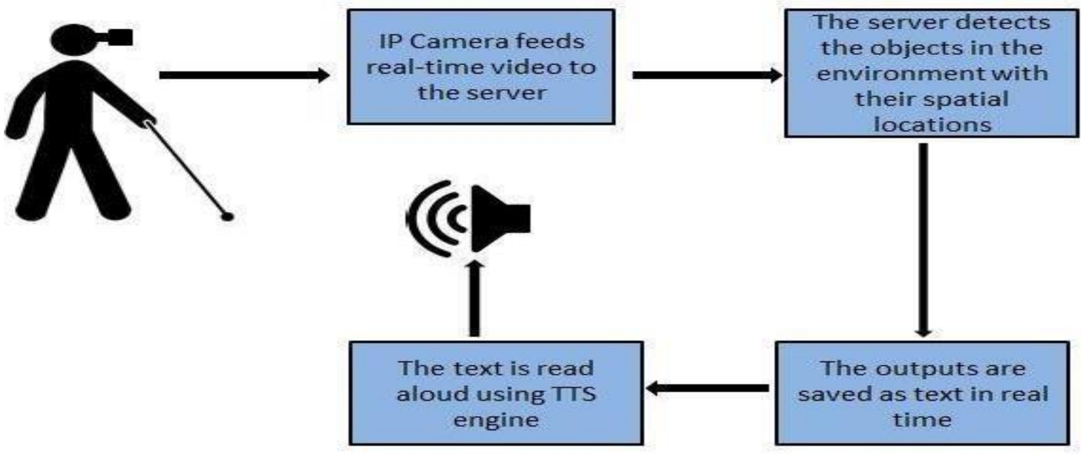
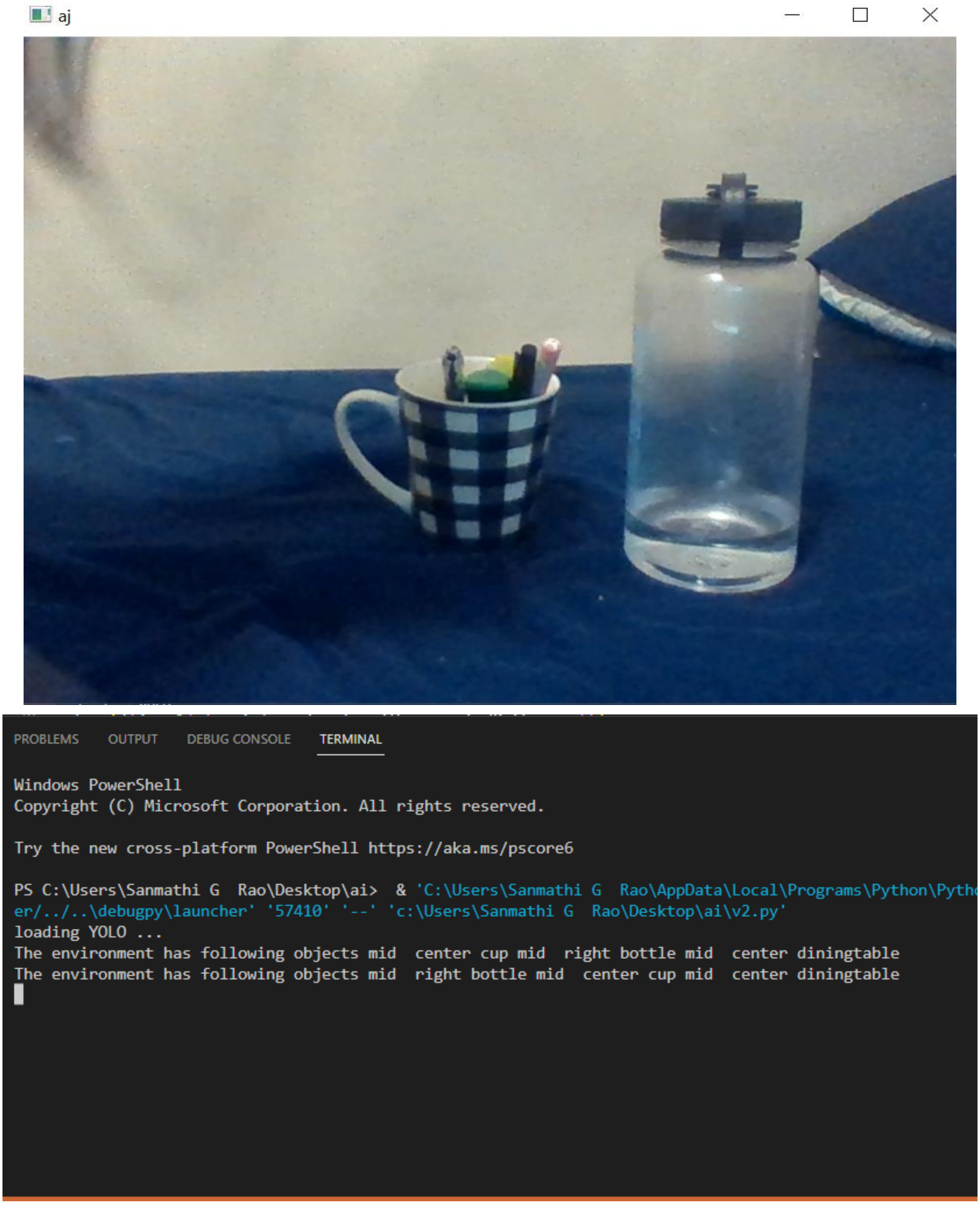

IP Camera feeds real-time video to the server
The server detects the objects in the environment with their spatial locations
The outputs are saved as text in real time
The text is read aloud using TTS engine

Fig 5.1.1: System work-flow diagram of Object detection



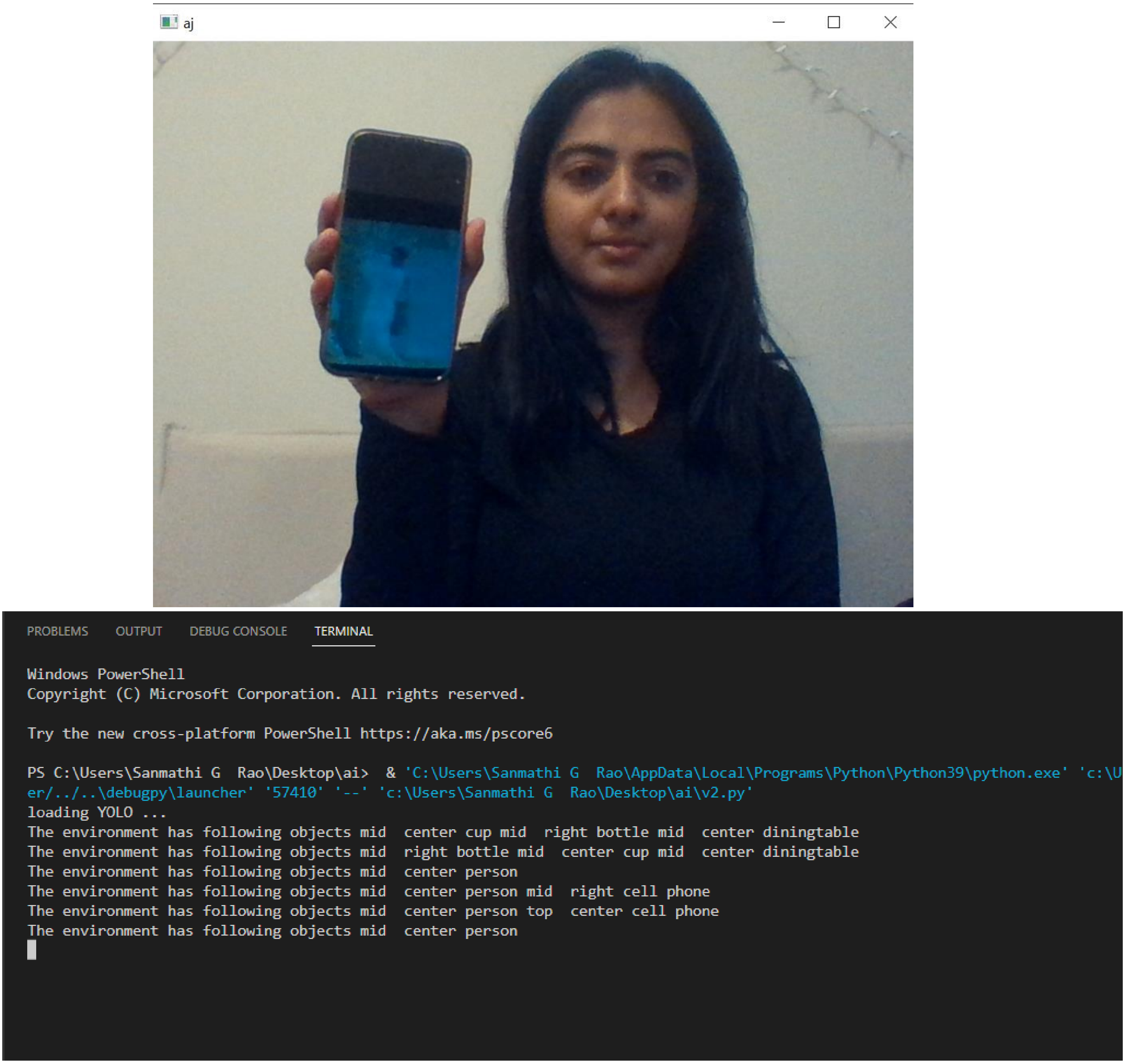
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Sanmathi G  Rao\Desktop\ai>  & 'C:\Users\Sanmathi G  Rao\AppData\Local\Programs\Python\Pytho
er/../..\debugpy\launcher' '57410' '--' 'c:\Users\Sanmathi G  Rao\Desktop\ai\v2.py'
loading YOLO ...
The environment has following objects mid  center cup mid  right bottle mid  center diningtable
The environment has following objects mid  right bottle mid  center cup mid  center diningtable
```

Fig 4.1.2: Object detection output-1

In the above output, we can observe that bottel and cup is detected with their correct spatial positions



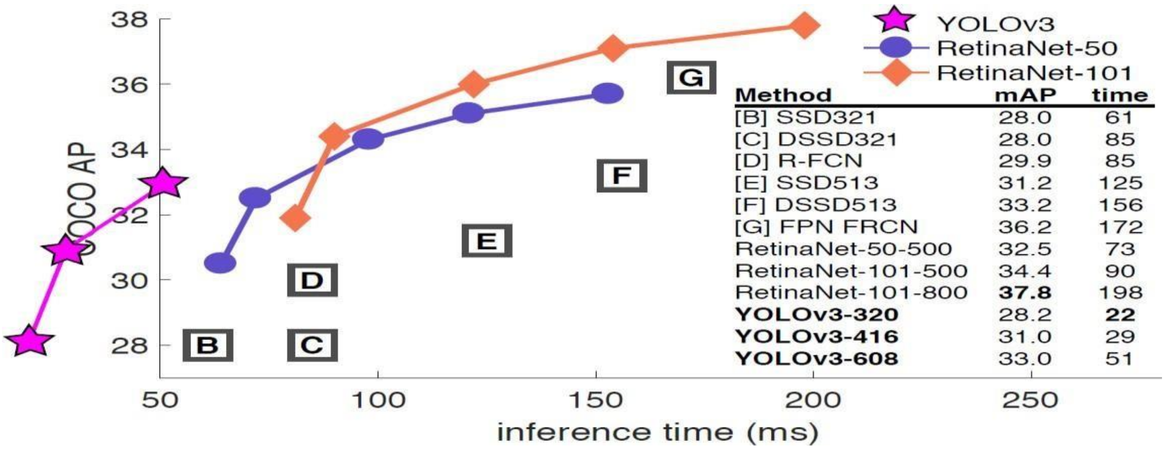
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Sanmathi G  Rao\Desktop\ai>  & 'C:\Users\Sanmathi G  Rao\AppData\Local\Programs\Python\Python39\python.exe' 'c:\U
er/../..\debugpy\launcher' '57410' '--' 'c:\Users\Sanmathi G  Rao\Desktop\ai\v2.py'
loading YOLO ...
The environment has following objects mid  center cup mid  right bottle mid  center diningtable
The environment has following objects mid  right bottle mid  center cup mid  center diningtable
The environment has following objects mid  center person
The environment has following objects mid  center person mid  right cell phone
The environment has following objects mid  center person top  center cell phone
The environment has following objects mid  center person
```

Fig 4.1.3: Object detection output-2

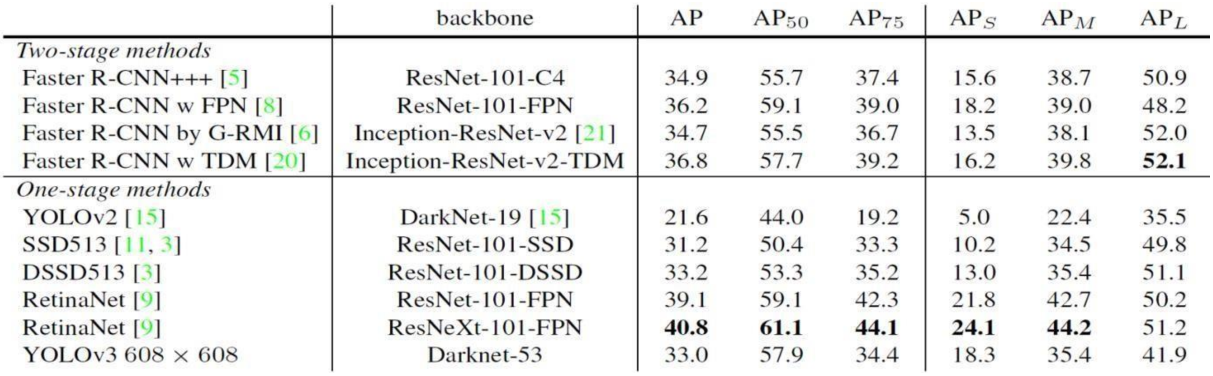
In the above output, we can observe that person and cell phone is detected with their correct spatial positions



Method	mAP	time
[B] SSD321	28.0	61
[C] DSSD321	28.0	85
[D] R-FCN	29.9	85
[E] SSD513	31.2	125
[F] DSSD513	33.2	156
[G] FPN FRCN	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	**37.8**	198
YOLOv3-320	28.2	**22**
YOLOv3-416	31.0	29
YOLOv3-608	33.0	51

Fig 4.1.4: Overall mAP from Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." *arXiv preprint arXiv:1804.02767* (2018).

In “overall mAP”, the YOLO v3 performance is decreased significantly. YOLO v3 -608, in51 milliseconds inference time got 33% of mAP. While RetinaNet-101-50-500 only got 32.5% mAP in 74 milliseconds inference time. Also YOLO v3 is 3-times faster with Single Shot Multibox Detector (SSD) versions.



	backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Two-stage methods							
Faster R-CNN+++ [5]	ResNet-101-C4	34.9	55.7	37.4	15.6	38.7	50.9
Faster R-CNN w FPN [8]	ResNet-101-FPN	36.2	59.1	39.0	18.2	39.0	48.2
Faster R-CNN by G-RMI [6]	Inception-ResNet-v2 [21]	34.7	55.5	36.7	13.5	38.1	52.0
Faster R-CNN w TDM [20]	Inception-ResNet-v2-TDM	36.8	57.7	39.2	16.2	39.8	**52.1**
One-stage methods							
YOLOv2 [15]	DarkNet-19 [15]	21.6	44.0	19.2	5.0	22.4	35.5
SSD513 [11, 3]	ResNet-101-SSD	31.2	50.4	33.3	10.2	34.5	49.8
DSSD513 [3]	ResNet-101-DSSD	33.2	53.3	35.2	13.0	35.4	51.1
RetinaNet [9]	ResNet-101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
RetinaNet [9]	ResNeXt-101-FPN	**40.8**	**61.1**	**44.1**	**24.1**	**44.2**	51.2
YOLOv3 608 × 608	Darknet-53	33.0	57.9	34.4	18.3	35.4	41.9

Fig 4.1.5: More details from Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." *arXiv preprint arXiv:1804.02767* (2018).

YOLOv3 is much preferred than SSD and has a similar performance as Deconvolutional-SSD. And it is found that YOLO v3 has better performance on AP-S but comparatively low performance on AP-M and AP-L. By using ResNet, FPN, G-RMI, and TDM, YOLO v3 has finer AP-S than 2-stage Fast R-CNN variants.

4.2OCR:

The fundamental block of this system is Tesseract and gtts, the below figure demonstrates the work-flow of the proposed system.

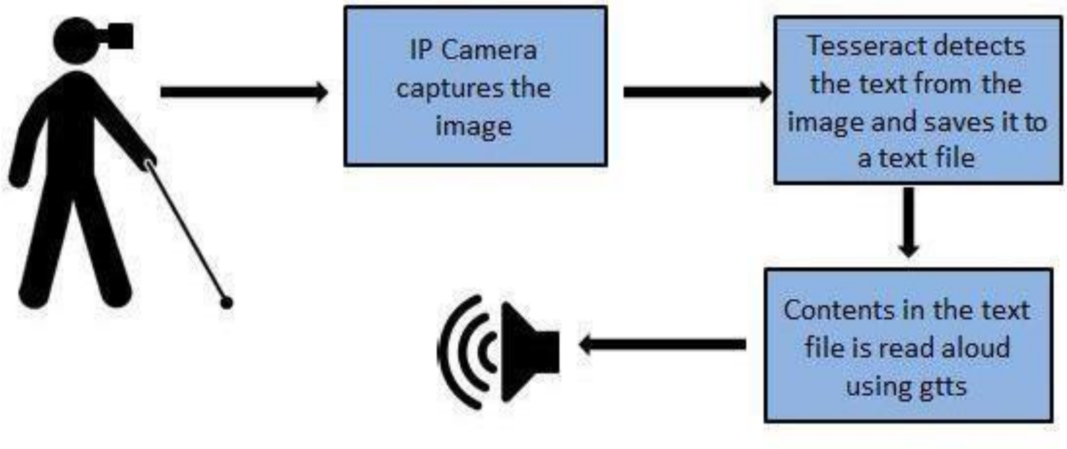
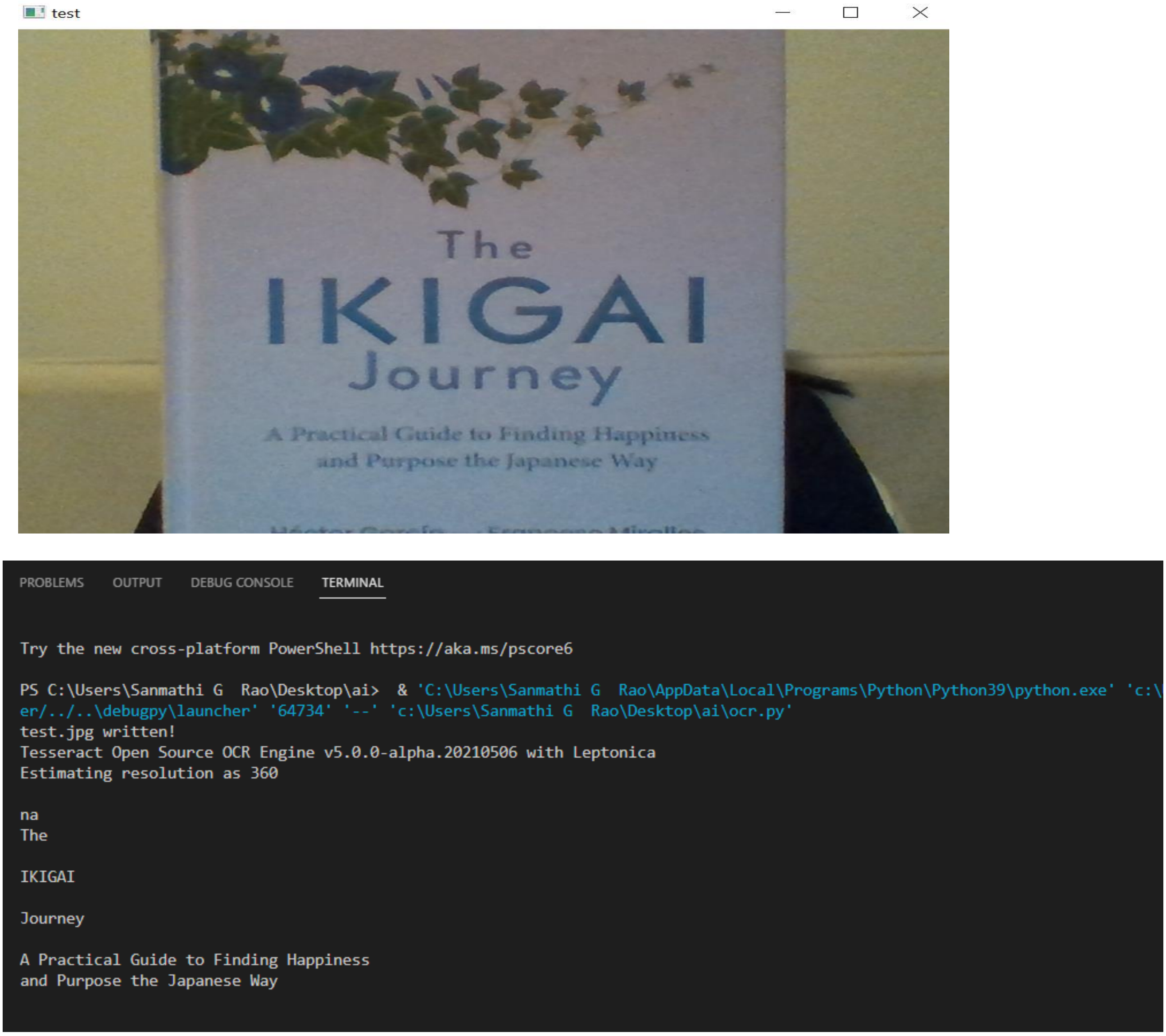

IP Camera captures the image
Tesseract detects the text from the image and saves it to a text file
Contents in the text file is read aloud using gtts

Fig 4.2.1: System work-flow diagram of OCR

The flow diagram of the OCR working in an experimental setup, where the live video captures the text images as input which is then fed to the server for processing, and the subject is assisted with audio output.


test
The
IKIGAI
Journey
A Practical Guide to Finding Happiness
and Purpose the Japanese Way
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL

```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Sanmathi G  Rao\Desktop\ai>  & 'C:\Users\Sanmathi G  Rao\AppData\Local\Programs\Python\Python39\python.exe' 'c:\
er/../..\debugpy\launcher' '64734' '--' 'c:\Users\Sanmathi G  Rao\Desktop\ai\ocr.py'
test.jpg written!
Tesseract Open Source OCR Engine v5.0.0-alpha.20210506 with Leptonica
Estimating resolution as 360

na
The

IKIGAI

Journey

A Practical Guide to Finding Happiness
and Purpose the Japanese Way
```

Fig 4.2.2: OCR output

In the above output we can observe that the text on the object i.e. from a book is detected with an accurate output.

CONCLUSION

The project presents a technique for helping people with visually impaired conditions. This project "Assistive Program for Blind People Using YOLO" is based on object detection, the program suggested has a basic architecture, which uses the YOLO V3 detection algorithm along with a pre-trained COCO dataset and gives viable audio feedback transforming the visual signal into a speech signal rendering it user friendly. Compared to the other algorithms the architecture in YOLOv3 is effectively enhanced and optimized. The OCR technology used for translation services does a hassle-free job of reading out any scanned PDF document, JPG, or PNG file with text. The desktop application framework can procure the benefits of any portable cameras. This project helps the person with visual impairment recognize the surrounding objects, identify the known persons, and also able to read any document. The preliminary tests show positive results as the consumer can easily distinguish the surrounding objects.

Future Scope

For the auxiliary advancement of our project, we would like to improve the experimental setup in a more convenient and easily accessible model for visually impaired people. We are also aiming on improvising YOLO and also to make OCR available in all Indian languages. Furthermore, we would like to deploy a web application to the server so that it is accessible by everyone in need. That will make our model work better and fulfill our goal. We would also like to incorporate other features to make it easier to use.

BIBLIOGRAPHY

1. Potdar, Kedar, Chinmay D. Pai, and Sukrut Akolkar. "A Convolutional Neural Network based Live Object Recognition System as Blind Aid." *arXiv preprint arXiv: 1811.10399* (2018).

2. Rahman, Ferdousi, Israt Jahan Ritun, and Nafisa Farhin. *Assisting the visually impaired people using image processing*. Diss. BRAC University, 2018.

3. Unnikrishnan, Ranjith, and Ray Smith. "Combined script and page orientation estimation using the tesseract OCR engine." *Proceedings of the international workshop on multilingual OCR*. 2009.

4. Smith, Ray. "An overview of the Tesseract OCR engine." *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007).* Vol. 2. IEEE, 2007.

Websites

[1] https://dzone.com/articles/understanding-object-detection-using-yolo

[2] https://www.pyimagesearch.com/category/object-detection

[3] https://medium.com/analytics-vidhya/yolo-v3-theory-explained-33100f6d193