Blind Vision

Karthik Kurella[1], Karthik Prasad[1], Pavan Karthik[1], Lokesh[1], Adithya Sai[1], Chaithanya[1] and Harini S[2]

[1]Department of MTech Integrated Software Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India.

[2]Associate Professor, Vellore Institute of Technology, Chennai, Tamil Nadu, India.

Abstract. Many people around the world cannot understand or realize the environment because of visual imparity. They may have difficulties in navigation, identifying objects which may lead to hinder their social behaviour. In the past, there had been other ways to deal with such challenges and daily routines. For the growing competitive world around, it is quite difficult for a visually impaired person to move around independently and identify surrounding objectives correctly with ease. Computer vision-based techniques have increased to the level to aid these real-world problems. Deep convolutional neural networks have developed very fast in recent years. It is very helpful to use computer vision-based techniques to help the visually impaired. This tool detects objects in an image and speaks the result to the user. We have included features for detecting text and summarising textual information. When the model is given with an image/scene it will decide which algorithm (Object detection / Text Summarisation) to be chosen. Switching between model's selection is done based on a static threshold value. All the results are read out for the user.

Keywords. YOLOv5, OCR, Text Summarization.

1. Introduction

In our daily life we come across many blind people who face many troubles in their daily life. We have designed a live detection system which helps blind people know text and interpret objects in their daily lives. To identify the desired letters in the label from the camera image, PyTesseract OCR (**Optical Character Recognition**) is used, if the words in the given image increase by a certain threshold value the recognised text will be shortened using the Text summarization technique and in order to identify the objects which are present in front of them the YOLOv5 is used. Once the system chooses among OCR and YOLO it reads out the particular scenario to the person holding the system. YOLO (**You Only Look Once**) is a state of art Object Detector which can perform object detection in real-time with a good accuracy. The major improvements include mosaic data augmentation and auto learning bounding box anchors. YOLOv5 is extremely fast and lightweight when compared to other YOLO versions and the accuracy is highly increased in order to know the exact objects present. OCR(Optical Character Recognition) helps in identifying the text present in the image once the number of words in the text is increased the text detected using OCR will be fed to text summarization which shortens the text and give us the output as a form of speech using text to speech conversion which helps the person know what's in front of them.

2. Background and Research

We conducted a survey by collecting 60 domain related articles and did research work on them. One of the popular methods for blind people to read information is Braille script. Though it is not much efficient, people still use the same for communication.

2.1 Existing System

We are giving a glance of some of the papers that we surveyed. One of which is ‘SMART SUMMARIZER FOR BLIND PEOPLE’ [1]. This paper focuses on sound rather than touch (Braille) and they keep track of important keywords to reduce the efforts of going through each line.

Another paper was ‘A CONVOLUTIONAL NEURAL NETWORK BASED LIVE OBJECT RECOGNITION SYSTEM AS BLIND AID’ [2], it was trained on ImageNet dataset for object recognition. The output was in the form of both audio and Braille text. This project was built using a pretrained neural network.

‘ENHANCED PORTABLE TEXT TO SPEECH CONVERTER FOR VISUALLY IMPAIRED’ [3]. This paper works for both handwritten text and printed text. For handwritten images, a scanner was used to scan the images (in an android phone) and it will be paired over Bluetooth. Then the text is extracted from the image using Tesseract and later converted to audio.

2.2 Proposed System

We have proposed a system which can dynamically change between Text recognition and Object Detection based on the input provided. We have integrated our model with a voice module, so that users can hear the summarised answer. Adapting to the change in the input scenario i.e., pipeline to the desired model and achieve the results. This level of adaptation is subjected to fixed thresholds in the model workflow (optimal value). Below is the total workflow of our proposed system.

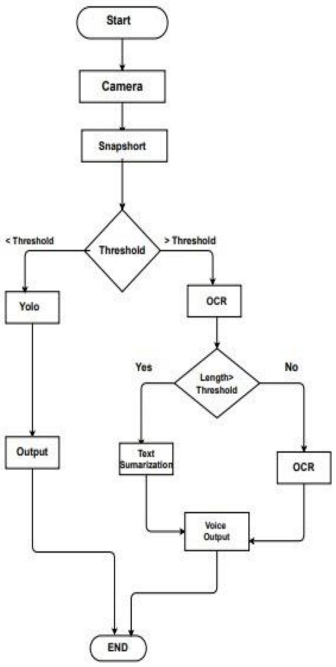


Figure 1 Architecture diagram

3. Methodologies

3.1 Object Detection

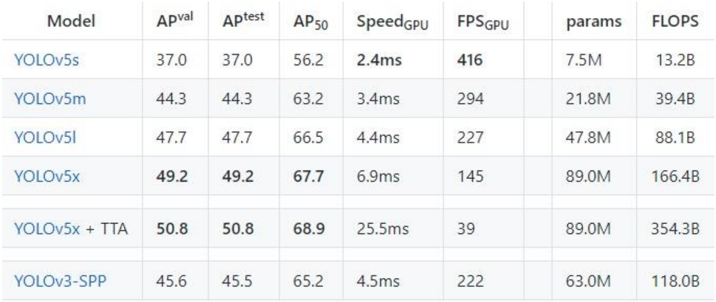
YOLO (**You only look Once**) is an DNN approach to identify the objects in the input image or video. This approach is a transfer-learning approach, prior on object detection has shown a significant accuracy. But in the recent years of innovation YOLO V5 has shown promising results in the detection of objects in a continuous changing environment. We are using this newly developed multi-model approach for the assisting the visually impaired persons in recognising objects in the surrounding environments. Unlike traditional recognition techniques which are solely built on static architecture. The YOLO V5 model has many versions that are built on same core functionality but with the tweaks on the performance edge to boost the output on the lower level or run time reduction (model training/validation). This YOLO model is inspired and built on version revision of the previous advancements of CNN. This YOLO model takes the footprint from the CNN architecture of the Google Net model which has 24 convolutional layers, attached to the two fully connected layers [4]. A final convolutional layer of this network can predict the simultaneous inputs using the various bounding boxes and mapping it to the associated object-detection probabilities.

Due to environmental changes and representations on the training / testing dataset of the YOLO framework, the framework may be less likely to return false or error detections when applied with peculiar occurrences in the real-world scenario.

YOLO has the following advantages (1) the processing time and is reduced, this model significance can have a ripple effect on the other modules work flow and initialization , the system is able to function at 60fps when coupled with the Nvidia 1050Ti GPU; (2) It takes the input as the image a compares with the heavily trained version of the most recent YOLO V5 (3) it is observed that this version of YOLO has the highest accuracy with lower number of false positives when compared with other state of art methods.

When it comes to other forms of detection of text, OCR this model is coupled with text summarisation and OCR recognition when no proper threshold levels of object are found in the image / scenario. The key innovation/refinement of the proposed approach consists in alternating between OCR and Text summarization with the threshold. On-fly this model continuously adapt to core feature changes of objects while dealing with noisy or other text occlusions that may cause the model to trigger other model with respect to threshold defined.

Table 1 Yolov5 Version Comparison [4]



Model	AP^{val}	AP^{test}	AP_{50}	Speed$_{GPU}$	FPS$_{GPU}$		params	FLOPS
YOLOv5s	37.0	37.0	56.2	**2.4ms**	**416**		7.5M	13.2B
YOLOv5m	44.3	44.3	63.2	3.4ms	294		21.8M	39.4B
YOLOv5l	47.7	47.7	66.5	4.4ms	227		47.8M	88.1B
YOLOv5x	**49.2**	**49.2**	**67.7**	6.9ms	145		89.0M	166.4B
YOLOv5x + TTA	**50.8**	**50.8**	**68.9**	25.5ms	39		89.0M	354.3B
YOLOv3-SPP	45.6	45.5	65.2	4.5ms	222		63.0M	118.0B

The above table shows the accuracies of all the existing versions of YOLOv5. v5x model gives higher accuracy with moderate run time. Though it requires a GPU to run, it gives prominent results. GPU is not a must because one can use Google Cloud to use YOLO for custom object detection.

3.2 Optical Character Recognition

Optical Character Recognition is a process of finding the text from an image provided the image can be captured from anywhere at any position. OCR also helps in identifying the handwritten characters also which can be converted into a perfect document after recognizing it with the help of OCR.

In our system we capture the image with the help of OpenCV then we implement the OCR module into the captured image.

Tesseract takes its input as a binary image and the outlines of the binary image are gathered completely with the help of nesting the collected outlines into those appropriate BLOBs (Binary Large Object),upon collecting the BLOBs these are correctly aligned into the lines and the lines are pre-processed with the help of noise reduction in the line by converting the image into grey scale image and the words are cut down based on the character spacing which are then categorized into fixed pitch and proportional text, fixed pitch are the group of words whose fonts are same in size and the proportional text are words whose font size are unequal. Upon classifying the lines which are in fixed pitch category will be cut down into separate words immediately and in case of proportional text the words are cut down based on their spaces like definite and fuzzy.

The spaces which are near to the given threshold value are converted into fuzzy so that the classification can be made at the end of the process. After cutting down the words these are put into priority and the one with least priority will be put aside and that will come into picture while classification. During classification the words which are identified will be compared to the dataset present and the words which are not correctly classified will be identified using the clustering from the dataset present. Upon completely performing all the steps the text will be completely extracted from the given image.

Then the identified text will be classified based on the word count, if word count exceeds the given threshold the text will go to text summarization, if it is less than threshold the text will be read out.

The system after running the program it will extract scores and geometrical data to help in the system and gets the score. The system will load the pre trained EAST text decoder, then it searches for the video input if it is not supplied it will open up the web cam and reads the frames and resizes it then and then it follows the algorithm defined to get the particular text from the inputted image

3.3 Text Summarization

The main **workflow** of this module is as follows:

Input document in a text format → check for sentence(cosine) similarity → weighing the sentences → selecting valuable sentences → desired output.

We are using extractive text summarization which tries to summarize an article by choosing the subset of the words and retain the most important sentences, this approach gives weights to the important part of sentences and with that sentences it will try to extract the summary, to extract the weights there are many techniques but we are using the cosine similarity to get the weights.

Initially we take input from the file and we read line by line and then we divide all the input data to the sentences and we will tokenize the words and we remove all the special characters and stop words and then we will extract the features from the text by using the cosine similarity between each sentence and then we construct the similarity score matrix for all the sentence and then the final step is to rank the sentences and Method will keep calling all other helper functions to keep our summarization pipeline going. Make sure to take a look at all the top sentences and that will be the output for text summarization, by this approach with less computation time we can achieve more accuracy so that we had chosen this approach

There are numerous content rundown strategies that will get the specific significance and it will require some investment If they have the less computational multifaceted nature and the precision of the content synopsis will be less and they utilize numerous profound learning procedures for text outline to get the exact importance. so, our strategy for text outline set aside extremely less effort for calculation and we likewise got the most precise importance for text rundown we need to integrate this entire module with the Optical character recognition (OCR). For summarizing the text, we require a base no of words i.e., 30 for satisfying this requirement we passed a condition that if words are greater than 30 it will dynamically implement text summarization with those words, and if the condition fails it will be implementing the text to speech module

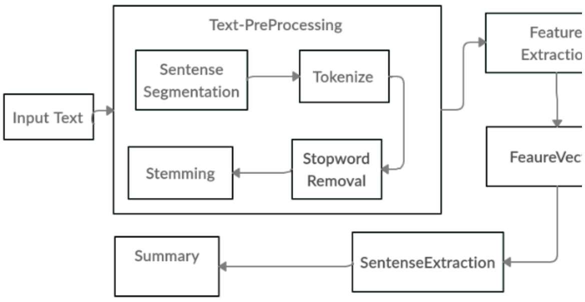
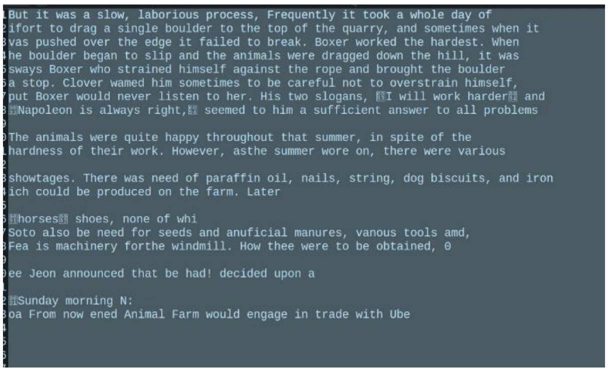


Figure 3 Text Summarization

Experiments and Results

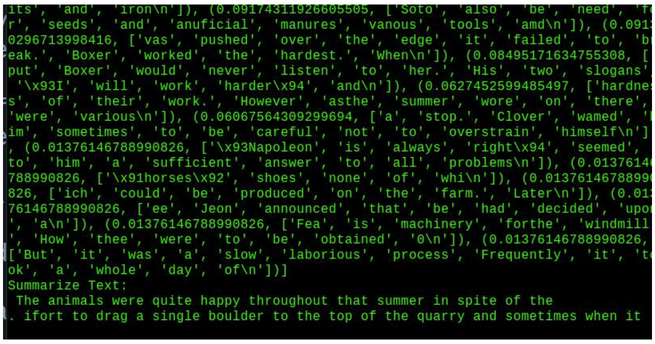


```
But it was a slow, laborious process, Frequently it took a whole day of
ifort to drag a single boulder to the top of the quarry, and sometimes when it
vas pushed over the edge it failed to break. Boxer worked the hardest. When
he boulder began to slip and the animals were dragged down the hill, it was
sways Boxer who strained himself against the rope and brought the boulder
a stop. Clover wamed him sometimes to be careful not to overstrain himself,
put Boxer would never listen to her. His two slogans, □I will work harder□ and
□Napoleon is always right,□ seemed to him a sufficient answer to all problems

The animals were quite happy throughout that summer, in spite of the
hardness of their work. However, asthe summer wore on, there were various

showtages. There was need of paraffin oil, nails, string, dog biscuits, and iron
ich could be produced on the farm. Later

□horses□ shoes, none of whi
Soto also be need for seeds and anuficial manures, vanous tools amd,
Fea is machinery forthe windmill. How thee were to be obtained, 0

ee Jeon announced that be had! decided upon a

□Sunday morning N:
oa From now ened Animal Farm would engage in trade with Ube
```

Figure 4 OCR Detected text from input

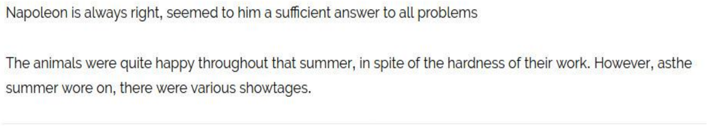
Text recognition and Summarization:

The provided screenshot is given when the module is tested separately. The input here is a novel shown in a webcam and our model has detected most of the content. Then this output was sent to the Summarization module. The summarised text was shown below.



```
its', 'and', 'iron\n']), (0.09174311926605505, ['Soto', 'also', 'be', 'need', 'fo
r', 'seeds', 'and', 'anuficial', 'manures', 'vanous', 'tools', 'amd\n']), (0.0913
0296713998416, ['vas', 'pushed', 'over', 'the', 'edge', 'it', 'failed', 'to', 'br
eak.', 'Boxer', 'worked', 'the', 'hardest.', 'When\n']), (0.08495171634755308, [
put', 'Boxer', 'would', 'never', 'listen', 'to', 'her.', 'His', 'two', 'slogans',
 '\x93I', 'will', 'work', 'harder\x94', 'and\n']), (0.0627452599485497, ['hardnes
s', 'of', 'their', 'work.', 'However', 'asthe', 'summer', 'wore', 'on', 'there',
'were', 'various\n']), (0.06067564309299694, ['a', 'stop.', 'Clover', 'wamed', '
im', 'sometimes', 'to', 'be', 'careful', 'not', 'to', 'overstrain', 'himself\n']
, (0.013761467889908026, ['\x93Napoleon', 'is', 'always', 'right\x94', 'seemed',
to', 'him', 'a', 'sufficient', 'answer', 'to', 'all', 'problems\n']), (0.0137614
788990826, ['\x91horses\x92', 'shoes', 'none', 'of', 'whi\n']), (0.01376146788990
826, ['ich', 'could', 'be', 'produced', 'on', 'the', 'farm.', 'Later\n']), (0.013
76146788990826, ['ee', 'Jeon', 'announced', 'that', 'be', 'had', 'decided', 'upon
', 'a\n']), (0.01376146788990826, ['Fea', 'is', 'machinery', 'forthe', 'windmill
', 'How', 'thee', 'were', 'to', 'be', 'obtained', '0\n']), (0.01376146788990826,
['But', 'it', 'was', 'a', 'slow', 'laborious', 'process', 'Frequently', 'it', 'to
ok', 'a', 'whole', 'day', 'of\n'])]
Summarize Text:
 The animals were quite happy throughout that summer in spite of the
. ifort to drag a single boulder to the top of the quarry and sometimes when it
```

Figure 5 Text Recognition and Summarization



Napoleon is always right, seemed to him a sufficient answer to all problems

The animals were quite happy throughout that summer, in spite of the hardness of their work. However, asthe summer wore on, there were various showtages.

Figure 6 Summarized output

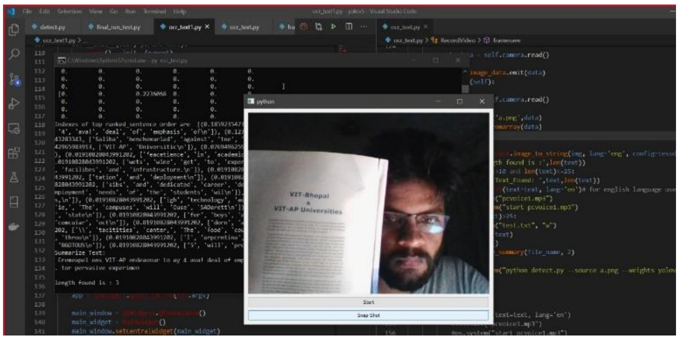


Figure7 Input image for the integrated system

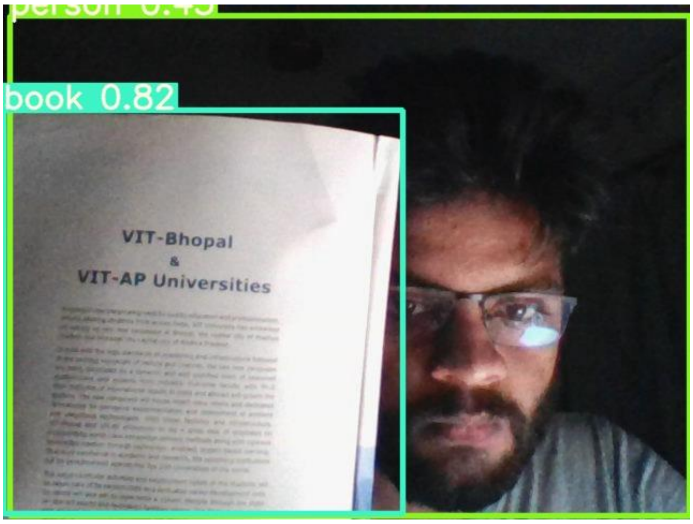
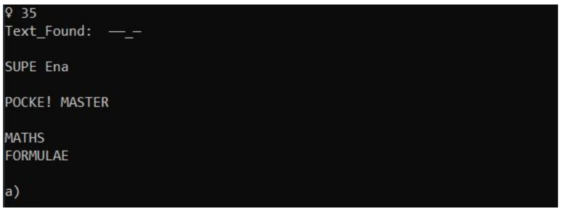


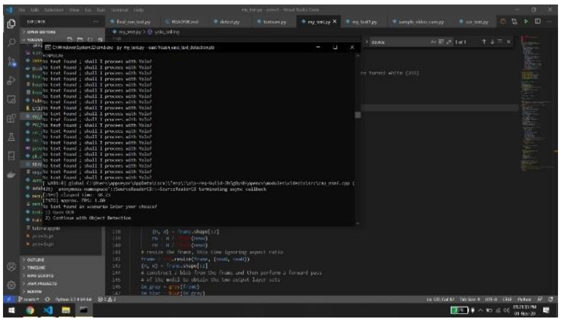
Figure 8 Output



```
9 35
Text_Found:  —_–

SUPE Ena

POCKE! MASTER

MATHS
FORMULAE

a)
```

Figure 9 OCR output after integration

The time took to produce output while testing with both OCR and object detection was around 90sec in a GPU system and also, we are not able to detect text from a live video. So, there is a necessity to upgrade our model to overcome the drawbacks. We have used EAST (Efficient and accurate scene text) detector for character recognition. EAST is a deep learning text detector. We have added custom layers to this network to boost the performance. After using EAST, we found that the model has boosted its performance and it is producing output only in 30secs. After integrating this with Object detection, time elapsed for detecting output was only 75sec. Results of this are shown below.



```
FORMULAE
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
No text found ; shall I procees with Yolo?
[ WARN:0] global C:\Users\appveyor\AppData\Local\Temp\1\pip-req-build-[illegible]\opencv\modules\videoio\src\cap_msmf.cpp (
435) `anonymous-namespace'::SourceReaderCB::~SourceReaderCB terminating async callback
[INFO] elapsed time: 38.23
[INFO] approx. FPS: 1.00
No text found in scenario Enter your choice?
 1) Open OCR
 2) Continue with Object Detection
```

Figure 10 Upgraded model

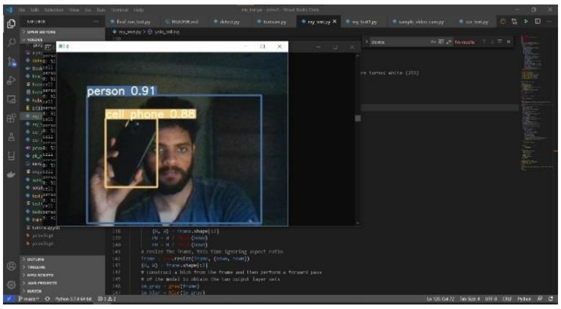


Figure 11 Obj Detec output

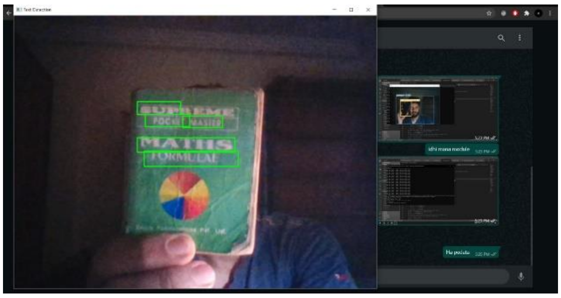
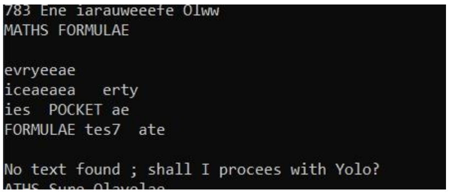


Figure 12 Live OCR implementation



```
783 Ene iarauweeefe Olww
MATHS FORMULAE

evryeeae
iceaeaea   erty
ies  POCKET ae
FORMULAE tes7  ate

No text found ; shall I procees with Yolo?
ATHS Supe Olavalao
```

Figure 13 OCR output

Conclusion

As specified in the Figure3, User will be executing this model in his/her system which enables webcam and if they wish to detect the object/text around then, they can select a snapshot option and click from a small interface built using PyQT5. The proposed system can switch based on the input given. If the model detects text more than certain threshold then we make it go to OCR else, it proceeds to Object detection. This system was mainly focussing on the words rather than sentences. So, there won’t be much scope for summarizing text here. Though we have upgraded the old text summarization module to handle escape characters such as' {'. As this was taking much time, we upgraded this to a reliable and efficient system took less time to produce output.

References

[1] Mona teja K, Mohan Sai.S, H S S S Raviteja D, Sai Kushagra P V 2018 SMART SUMMARIZER FOR BLIND PEOPLE. *3rd International Conference on Inventive Computation Technologies (ICICT)*

[2] Kedar Podtar, Chinmay Pai, Sukrut Akolkar 2018 A Convolutional Neural Network based Live Object Recognition System as Blind Aid. arXiv:1811.10399 **[cs.CV]**

[3] Chithra Selvaraj, Bhalaji Natarajan 2018 Enhanced portable text to speech converter for visually impaired. *International Journal of Intelligent Systems Technologies and Applications*

[4] Glen Jocher 2020 YoloV5 https://github.com/ultralytics/yolov5

[5] Jarlin James, Michael Aldo, Adellina Andrew 2018 Image Text to Speech Conversion Using OCR Technique in Raspberry PI. *International Journal of Scientific & Engineering Research*

[6] Pranab Das, Kakali Acharjee 2015 VOICE RECOGNITION SYSTEM: SPEECH-TO-TEXT. *Journal of Applied and Fundamental Sciences*

[7] Xiaodong Yang and YingLi Tian Chucai Yi, Aries Arditi 2010 Context-based indoor object detection as an aid to blind persons accessing unfamiliar environments. *Proceedings of the 18th International Conference on Multimedea 2010, Firenze, Italy*