Divya-Drishti: An Independent Aid for the Visually Impaired

Jay Jhaveri
Computer Engineering
Vivekanand Education Society's
Institute of Technology
Mumbai, India
2018.jay.jhaveri@ves.ac.in

Prem Chhabria
Computer Engineering
Vivekanand Education Society's
Institute of Technology
Mumbai, India
2018.prem.chhabria@ves.ac.in

Dr. Mrs. Sharmila Sengupta
Professor
Vivekanand Education Society's
Institute of Technology
Mumbai, India
sharmila.sengupta@ves.ac.in

Abhay Gupta
Computer Engineering
Vivekanand Education Society's
Institute of Technology
Mumbai, India
2018.abhay.gupta@ves.ac.in

Neeraj Ochani
Computer Engineering
Vivekanand Education Society's
Institute of Technology
Mumbai, India
2018.neeraj.ochani@ves.ac.in

***Abstract*—The Objective of this system is to help/guide the visually challenged people with the help of a smart device using an Android Phone. This device will help the visually challenged person to get greater sense of awareness of surroundings around him/her and will also help him/her by protecting them against frauds. What makes this device innovative is that the device is completely Internet Free and also helps in effective communication with the help of Voice Commands as this is the only medium through which a visually challenged person can effectively communicate with an external device. Our device has successfully managed to implement multiple daily usage features like OCR, Bill Reading, Text Summarization, Mask Detection, Color Detection, Currency Detection in a single standalone portable system.**

***Keywords*—Blind, Single Board Computer (Raspberry Pi), Android, Internet of Things, Covid 19, TTS, OCR**

I. Introduction

In today's cruel world Blind people face many quandaries in everyday mundane activities. These may include money handling, newspaper reading, bill reading, rudimental understanding of one's environment. They require a Braille type of paper to read and understand the things customary humans take for granted. It would be consequential to them if these messages were to be integrated everywhere for them to lead a normal life.

At the moment our world is in the middle of a crisis, the Covid 19 Pandemic. We have to maintain a six-foot distance among us, merged with the utilization of masks. This too is now added to the already long list of problems faced by visually impaired people (VIP).

We endeavor to solve these lists of quandaries by implementing an accumulation of different technologies.

The Supervised Image Classification Algorithm helps in detecting money and day-to-day objects. We withal use it to differentiate humans with and without masks.

The Optical Character Recognition (OCR) helps in recognizing the texts from the image. It is widely used in documents to convert them into electronic copies which can be edited according to the user [1].

The verbalization synthesis (TTS) technology engenders digital format of output for the text apperceived from the image into the audio that will avail the blind people. The conversion of text into audio is able to convey the messages in the native language of the verbalizer. The recent development in this field avails in engendering a voice that impeccably syncs with the inchoation of the verbalizer [2]. This is an efficient way of conveying the messages to those people.

A system can be defined in such a way that VIPs can have an efficient way of interaction with the system so that the messages can be conveyed facilely. This system uses camera for capturing the image or picture by the embedded camera on the smartphone. After obtaining the picture, the information i.e. the output is processed by utilizing TTS module and hence the text information is converted into verbalization by the audio system.

The principal of the phone is that to have an interactive system to capture the image through image acquisition and the audio form of output is obtained through the TTS module.

II. Lacuna in the existing system

A. *Internet traffic*

All the existing systems needs an active internet connection to make API calls to the cloud. This is an astronomically immense quandary while accommodating the VIPs, as this adds adscititious variables out of the VIP's control (Microsoft Seeing AI) [3].

B. *Exorbitant prices*

Existing systems charge an astronomical amount which is even not affordable to people with eyes let alone VIPs (OrCam) [4]. Others charge a monthly/ annual subscription [5].

C. *Lacking Cross Platform capabilities*

Existing solutions are platform-dependent and do not work on multiple operating systems [3]. They also fail to run on low-end processor phones.

D. *Non-Blind Friendly UI*

The graphical user interface of free applications is not at all convenient to a VIP [3].

E. *Standalone Products*

Existing products do not provide an ALL-IN-ONE solution to combat all the major problems faced by a VIP [6].

III. Proposed System Architecture

Fig. 1 shows the detailed architecture of our independent aid for VIPs system. The entire product consists of two main divisions:

- Server (Raspberry Pi)
- Client (Smart Phone)

Being fully wireless, it is mandatory that both the server and the client should be connected on the same network. Server follows a static IP design, hence being capable of handling change of clients.

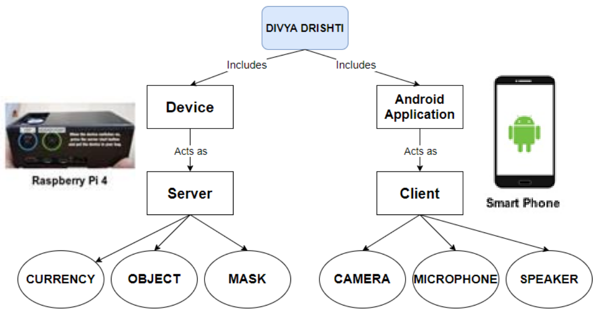


Fig. 1. System Architecture

A. Server

The server is hosted on a Raspberry Pi 4 module. The server is the main backbone of our product. It enables the whole system to be portable and hence even supports cross-platform usage.

We have researched on multiple single-board computers.

- Arduino UNO and Raspberry Pi ZERO’s processors were not up to the mark required by our product.
- Raspberry Pi 3, though capable of handling the image processing still gave a latency of 5-6 seconds per command which is not acceptable in the real world.
- Jeston Nano, albeit having excellent processors is exorbitantly priced and needs to be imported which is not practical.
- Determinately we concurred to utilize Raspberry Pi 4 module, which is the best of both worlds.

B. Client

The Client consists of a smartphone. A smartphone is used to provide an interactive interface between a VIP and our server. An application is installed on the phone, which automatically connects to the server after a one-time initial setup.

The Divya Drishti android application requires the mobile hotspot to be switched on. Once the hotspot is switched on, the device (server) is started and it gets automatically connected to the mobile's hotspot. Note that the internet need not be switched on. Once the device (server) is connected to the phone, the user gets an audio feedback stating that the server is now connected and is ready to function. The connection between the device and the phone obeys the TCP/IP networking protocol. The Divya Drishti android application has just one screen containing only the camera preview. There are no buttons, no menu bars, no tool-bars for interaction. The interaction between the user and the application is solely based on single and double screen-taps and voice input!!

IV. System Design and Working

A. Hardware Design

The System is designed as a partnership system between the server and a client. The server consists of parts as follows:

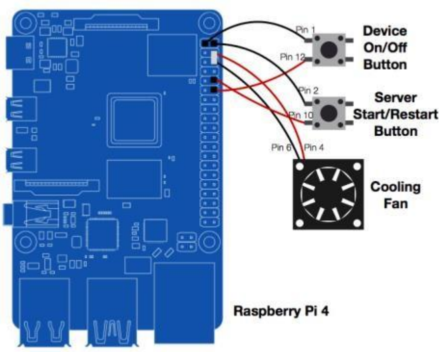


Fig. 2. Connections

1) All the connections are shown in Fig. 2. The server mainly consists of a Raspberry Pi unit. It is further fortified by a Cooling Fan and powered by a Lithium battery power pack.

2) The user interface for user-server consists of two buttons. First button is utilized to start up and shut down the system. Second button is to be pressed once to start the server program. A second press of the button, restarts the server.

3) The whole server is enclosed into a compact 3D model and the end product along with the client is shown in Fig 3.

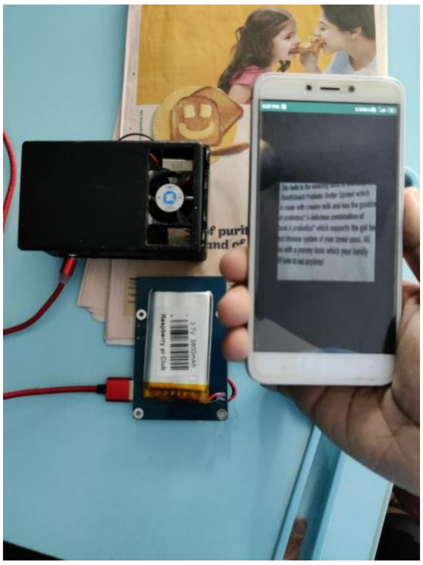


Fig 3. Divya-Drishti End Product

The client consists of a smartphone:

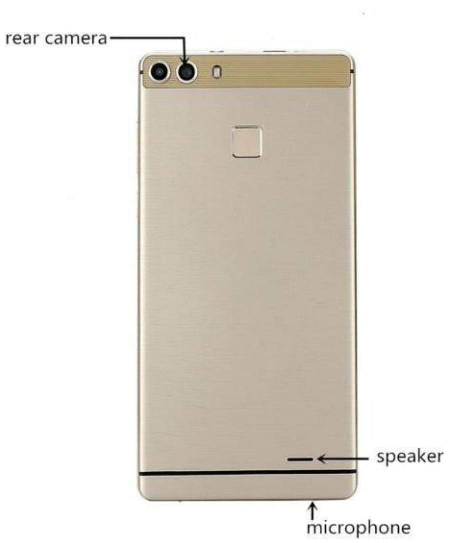


Fig. 4 Client

1) Fig. 4 shows a depiction of a general client system. The smartphone's rear camera is used to connect our server to the real world via the medium of multimedia.

2) The microphone is utilized to take voice commands from the user and deliver the command to the server. The speaker is used to deliver the output in audio format to a VIP.

B. Software Design

Fig 5. shows detailed description of our software design.

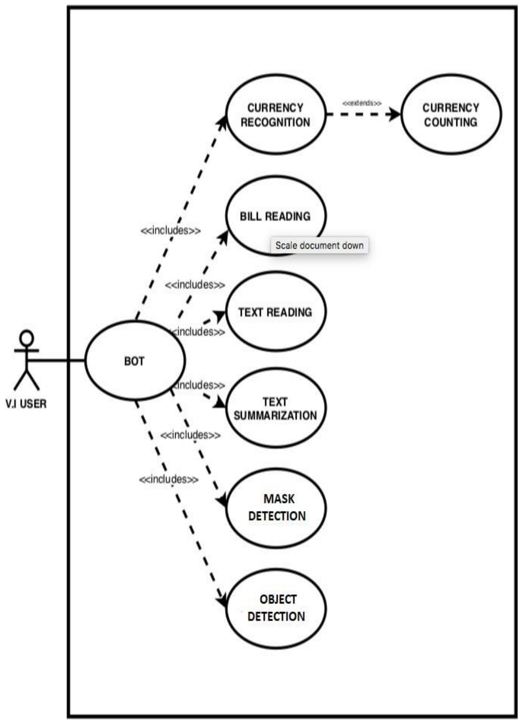


Fig. 5 Software Design

Our software is bifurcated into two parts, namely:

a) Android app: The app is developed using Android Java. It is a straightforward app with minimalistic UI design as shown in Fig. 6.

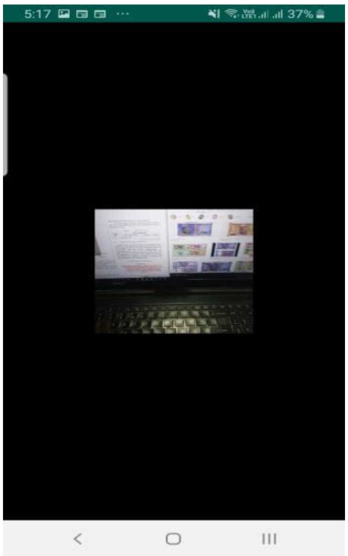


Fig 6. App’s UI

The UI of the app only comprises of a single camera preview window as shown in Fig. 6. It works on haptic feedback from the user. One single tap anywhere on screen captures an image and waits for a voice command by the user. It then sends the image captured along with the voice command to the server and responds with the appropriate message/ output. Double tap on screen is used to cancel the current ongoing operation in case of a user mistake.

b) Python: Python is the base language of our server. It receives the above-mentioned images and commands and reciprocates with an appropriate answer. Everything here is done offline without the need of any kind of internet connection. We have adopted a bottom up approach, this enables us to quickly add more functionalities to the product without any invasive procedures. The smaller modules are explained in detail in the Methodology section of our paper.

V. Methodology

A. Currency Recognition

We have used “864*864” dataset [7], which contained 6853 unique images, to train an image classification model on the GCP console using Google’s Vision API. A VIP clicks a photo of a note, which is sent to the server along with voice command of single detection or counting of notes. The basic structure of the module is shown in Fig 7.

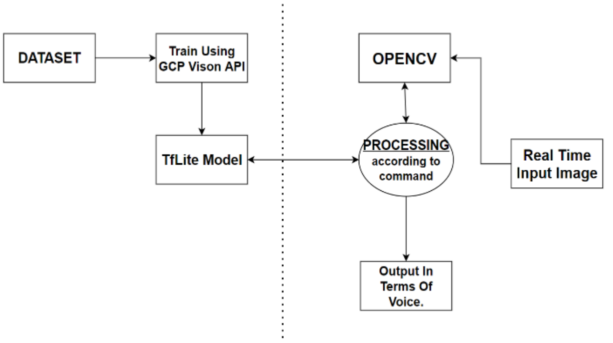

DATASET
Train Using GCP Vison API
TfLite Model
OPENCV
PROCESSING according to command
Real Time Input Image
Output In Terms Of Voice.

Fig. 7 Structure of module

B. *Mask Detection*

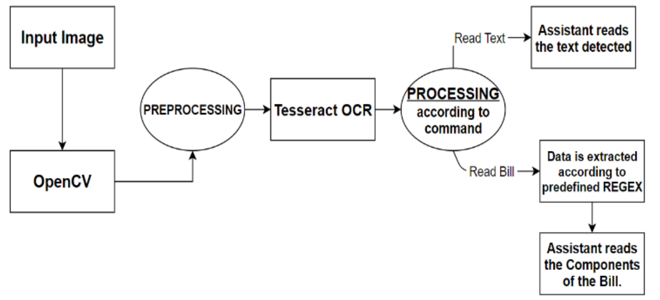
We have used the Real-world masked face recognition dataset [8]. After cleaning and labeling, it contains 5,000 masked faces of 525 people and 90,000 normal faces. Its structure is similar to the structure of the currency module Fig 7. It is able to differentiate between a person who is wearing a mask or not. It even classifies a burkha/ handkerchief as NOT a mask. It also returns NOT a mask if a person is not covering their nose and mouth properly with a mask.

C. *Object Detection*

We have currently used pretrained models of everyday objects over the coco database [9]. This helps a VIP to detect and find small objects. Its structure is also similar to the currency module Fig 7.

D. *Text Reading*

This module uses python's Pytesseract library to extract the text from the input image. Pytesseract is an open-source OCR (Optical Character Recognition) engine Fig 8.



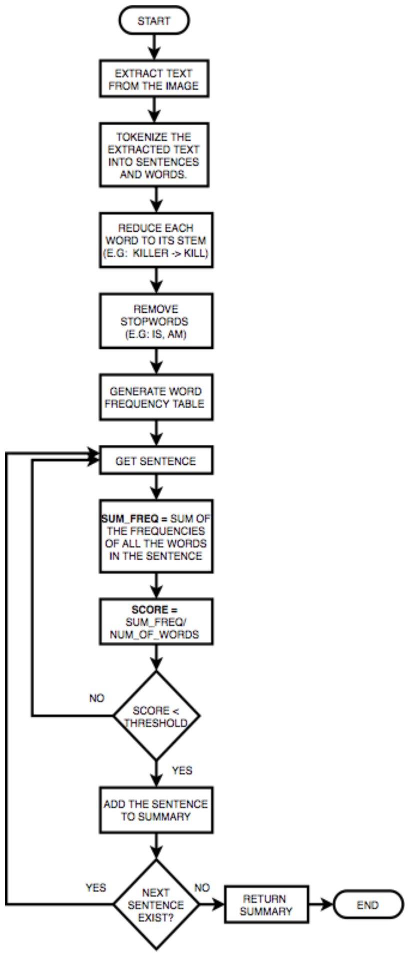
Input Image
OpenCV
PREPROCESSING
Tesseract OCR
PROCESSING according to command
Read Text
Assistant reads the text detected
Read Bill
Data is extracted according to predefined REGEX
Assistant reads the Components of the Bill.

Fig. 8 Text Reading

However, it comes with a limitation that the orientation of the text should be proper enough for its detection. The system does pre-processing on the extracted text to determine the count of words that actually make sense (words which are present in the dictionary). If this count, exceeds the system-defined threshold, then the text is accepted, else the user is given a prompt to reorient the text.

E. *Text Summarization*

The Basic Text Reading module returns the text extracted from the input image. The general text summarization algorithm is then applied on this extracted text to give a succinct output. This feature is useful when the user doesn't want the system to read the entire text, rather wants to know just the essence of it. Given the input, the text is sentence-tokenized and word-tokenized followed by stopword removal (removal of words like a, an, is, the, etc). A frequency table is then generated from the preprocessed text which stores the frequency of every word in it.



START
EXTRACT TEXT FROM THE IMAGE
TOKENIZE THE EXTRACTED TEXT INTO SENTENCES AND WORDS.
REDUCE EACH WORD TO ITS STEM (E.G: KILLER -> KILL)
REMOVE STOPWORDS (E.G: IS, AM)
GENERATE WORD FREQUENCY TABLE
GET SENTENCE
SUM_FREQ = SUM OF THE FREQUENCIES OF ALL THE WORDS IN THE SENTENCE
SCORE = SUM_FREQ/ NUM_OF_WORDS
NO
SCORE < THRESHOLD
YES
ADD THE SENTENCE TO SUMMARY
YES
NEXT SENTENCE EXIST?
NO
RETURN SUMMARY
END

Fig. 9 Text Summarizer

The frequency table generation is then followed by sentence scoring wherein each sentence in the input text is assigned a score. For every word in the sentence, its frequency is added together. The resulting sum from all of the words in that sentence is then normalized by dividing it with the word count which results in the sentence score. Sentences with scores lesser than a system-defined threshold are eliminated, while those which exceed it are included in the resultant summary.

F. *Bill Reading*

The Basic Text Reading module Fig. 8 returns the text extracted from the input image. Upon receiving the text, the python's regex (Regular Expressions) library converts the plain text into records split by a new line. If the number of records with FOUR floating point numbers (Qty, MRP, Rate, Amount) exceeds the system-defined threshold, then the image can be considered to be a supermarket bill and can be processed further. Once the system determines that the image is of a supermarket bill it applies various permutations (4P2) and checks if the multiplication of two floating point numbers matches one of the four floating point numbers. Based on this, the module finds the mapping of the four floating point numbers to Qty, MRP, Rate and Amount. The remaining text apart from these 4 numbers is considered to be the product name.

VI. Result of implementation

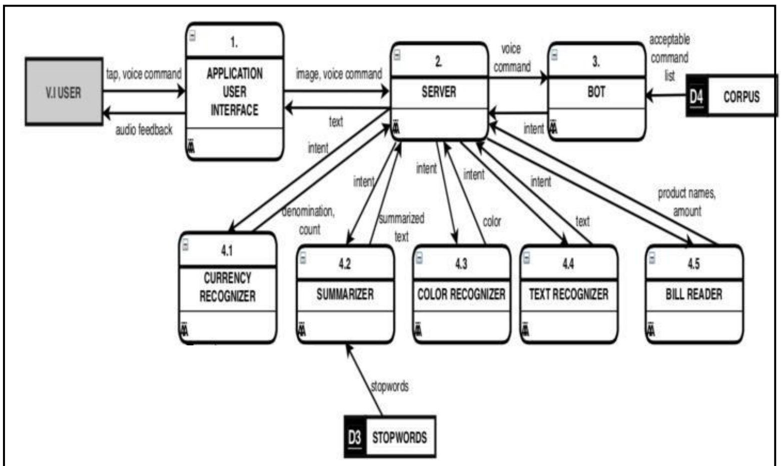
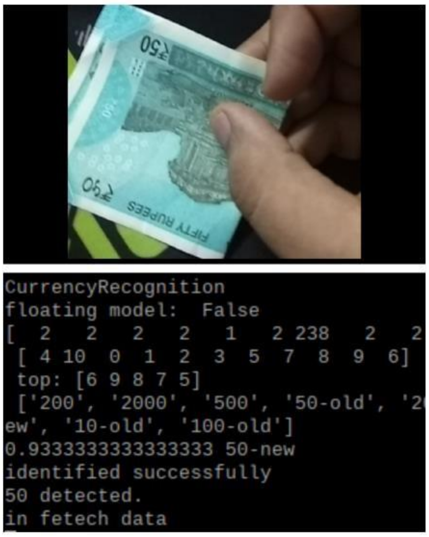


Fig. 10 depicts the final model of our whole system.

A. *Currency Recognition*

Fig. 11 shows the input and output results of a 50 Rs. note.

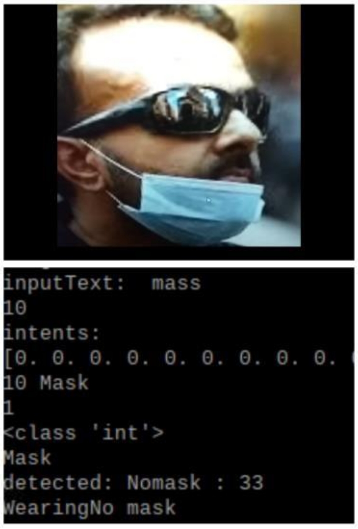


```
CurrencyRecognition
floating model:  False
[  2   2   2   2   1   2 238   2   2
 [ 4 10  0  1  2  3  5  7  8  9  6]
 top: [6 9 8 7 5]
 ['200', '2000', '500', '50-old', '2
ew', '10-old', '100-old']
0.9333333333333333 50-new
identified successfully
50 detected.
in fetech data
```

Fig 11. Currency Recognition

B. *Mask Detection*

Fig. 12 shows the identification of a wrongly positioned mask on a person.

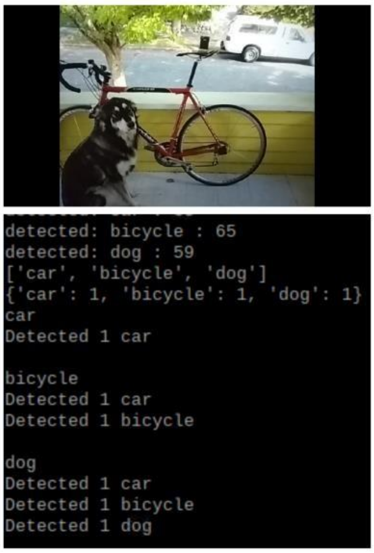


```
inputText:  mass
10
intents:
[0. 0. 0. 0. 0. 0. 0. 0. 0.
10 Mask
1
<class 'int'>
Mask
detected: Nomask : 33
WearingNo mask
```

Fig 12. Mask Detection

C. *Object Detection*

Fig. 13 shows multiple objects detected by our module. We concluded that it needs more work to be done. We also need to add more classes for the module to detect.

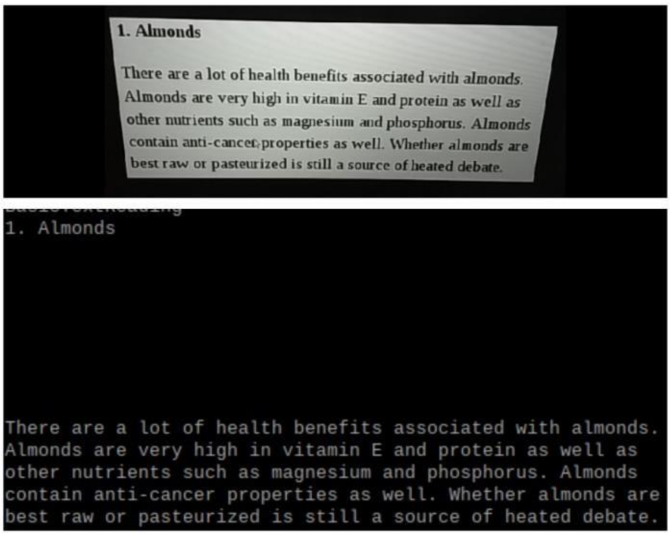


```
detected: bicycle : 65
detected: dog : 59
['car', 'bicycle', 'dog']
{'car': 1, 'bicycle': 1, 'dog': 1}
car
Detected 1 car

bicycle
Detected 1 car
Detected 1 bicycle

dog
Detected 1 car
Detected 1 bicycle
Detected 1 dog
```

Fig 13. Object Detection

D. *Text Detection*

Fig. 14 shows the results of the fully offline OCR of our module on the given image. From multiple testing and iterations, we conjectured that to get definitive and accurate results from our OCR module, one would need adequate amounts of lightning conditions.



1. Almonds

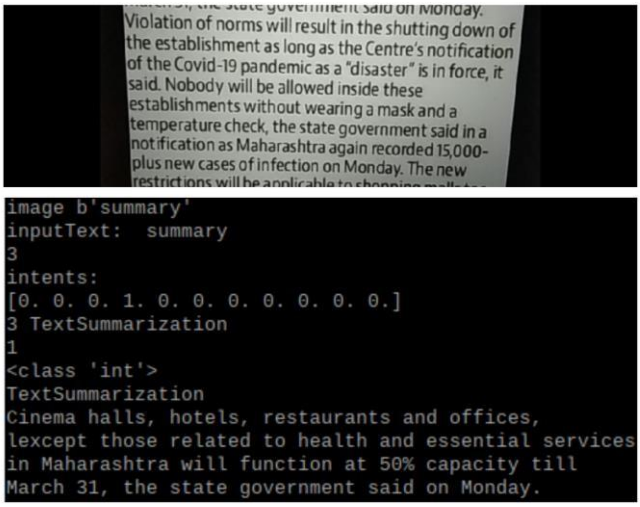
There are a lot of health benefits associated with almonds. Almonds are very high in vitamin E and protein as well as other nutrients such as magnesium and phosphorus. Almonds contain anti-cancer properties as well. Whether almonds are best raw or pasteurized is still a source of heated debate.

```
1. Almonds




There are a lot of health benefits associated with almonds.
Almonds are very high in vitamin E and protein as well as
other nutrients such as magnesium and phosphorus. Almonds
contain anti-cancer properties as well. Whether almonds are
best raw or pasteurized is still a source of heated debate.
```

Fig. 14 Text Reader

E. *Text Summarization*

Fig. 15 shows the results of the summarization of text using NLTK and tokenization. Due to summarization being dependent on Text Detection module as mentioned before, it still has flaws to be solved.



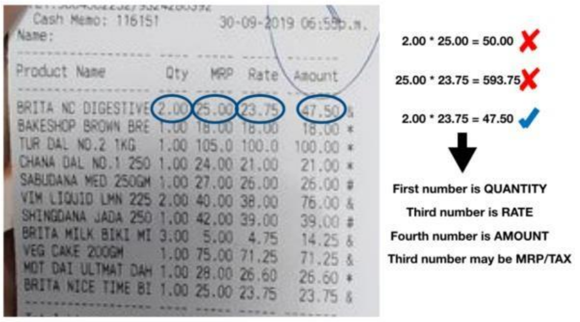
Violation of norms will result in the shutting down of the establishment as long as the Centre's notification of the Covid-19 pandemic as a "disaster" is in force, it said. Nobody will be allowed inside these establishments without wearing a mask and a temperature check, the state government said in a notification as Maharashtra again recorded 15,000-plus new cases of infection on Monday. The new

```
image b'summary'
inputText:  summary
3
intents:
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0.]
3 TextSummarization
1
<class 'int'>
TextSummarization
Cinema halls, hotels, restaurants and offices,
lexcept those related to health and essential services
in Maharashtra will function at 50% capacity till
March 31, the state government said on Monday.
```

Fig. 15 Text Summarizer

F. *Bill Reading*

Fig. 16 shows the results of bill reading using our module. Being dependent on Text recognition, it still has multiple loop holes to be worked upon. The case is not helped by various formats of bills out there in the world.



Cash Memo: 116151 30-09-2019 06:55p.m.
Name:

Product Name	Qty	MRP	Rate	Amount	
BRITA NC DIGESTIVE	2.00	25.00	23.75	47.50	&
BAKESHOP BROWN BRE	1.00	18.00	18.00	18.00	*
TUR DAL NO.2 1KG	1.00	105.0	100.0	100.00	*
CHANA DAL NO.1 250	1.00	24.00	21.00	21.00	*
SABUDANA MED 250GM	1.00	27.00	26.00	26.00	#
VIM LIQUID LMN 225	2.00	40.00	38.00	76.00	&
SHINGDANA JADA 250	1.00	42.00	39.00	39.00	#
BRITA MILK BIKI MI	3.00	5.00	4.75	14.25	&
VEG CAKE 200GM	1.00	75.00	71.25	71.25	&
MDT DAI ULTMAT DAH	1.00	28.00	26.60	26.60	*
BRITA NICE TIME BI	1.00	25.00	23.75	23.75	&

Fig. 16 Bill Reader

All the output is spoken aloud in clear English and Devanagari language to the VIP user, utilizing the phone's speaker system.

This device has the potential to completely revamp a VIP's Life. It is to be considered the first step towards independence of visually impaired people.

After multiple iterations of debugging and reviews, the project can still be worked upon. We consider the project yet as an Alpha build. Our code can be found on our GitHub page [10].

VII. Future scope

1) Use of Solar panels to make the device sustainable.

2) Use of Jetson Nano and a high-end camera to make the device completely independent.

3) Document classification will eliminate the need for the user to give voice commands in order to select the module to be called.

4) Voice-based calculator can be an add-on.

5) Semantics in text recognition can be captured.

6) Increase OCR accuracy in different lightning conditions and improve the Information extraction accuracy of bills.

VIII. Conclusion

The positive impact of the proposed system can change lives. It can engender confidence and knowledge of self-worth which is critical to making a cogent difference in the total quality of life for blind people.

It will open the way for the blind to participate in recreational and social activities with the sighted community and will give blind people the confidence and motivation to learn skills and expertise that will better enable them to maintain gainful employment.

We hope this simple device will indeed one day help a person in need.

References

[1] ELMORE, M. AND MARTONOSI, M. "A morphological image Preprocessing suite for OCR on natural scene images", 2008

[2] THOMAS, S. "Natural Sounding Text-To-Speech Synthesis Based On Syllable-Like Units, Department Of Computer Science And Engineering Indian Institute Of Technology Madras", 2007.

[3] Seeing Ai requires net, Only available on IOS, Non-Blind-Friendly UI: https://www.microsoft.com/en-us/ai/seeing-ai

[4] OR cam-2.5 Lakhs: https://www.orcam.com/en/

[5] Aira-High Monthly subscription cost: https://aira.io/pricing

[6] Roshni App-Only Currency Recognition : https://www.digit.in/news/apps/roshni-an-android-app-to-help-the-visually-impaired-recognize-currency-notes-46026.html

[7] Currency Dataset: https://github.com/Jaydeep-Chaudhary/Indian-currency-notes_dataset

[8] Mask Dataset: https://github.com/X-zhangyang/Real-World-Masked-Face-Dataset

[9] Coco Dataset: https://cocodataset.org/

[10] Code: https://github.com/JayJhaveri1906/Divya-Drishti