IJSRTM
https://www.ijsrtm.com
Vol.3 Issue 2 June 2023: 12-19
Published online 11 June 2023
E-ISSN: 2583-7141

International Journal of Scientific Research in Technology & Management

Dual Purpose Autonomous Self Learning based Training Device for People with Alzheimer and Visually Challenged

J. Hemavathy
Dept. of Information Technology,
Panimalar Engineering College
Chennai, India

A. Akshaya
Dept. of Information Technology,
Panimalar Engineering College
Chennai, India

P.K. Narmadha
Dept. of Information Technology,
Panimalar Engineering College
Chennai, India

K. Nithya
Dept. of Information Technology,
Panimalar Engineering College
Chennai, India

***Abstract*—Alzheimer's disease is a progressive brain disorder, irreversible that slowly destroys memory and thinking skills. Recent studies have demonstrated that potential of visual interventions to improve the functioning of AD patients. Therefore, clarification of visual deficits in AD and possible mechanisms underlying these deficits are needed. We Propose a camera-based detection for visually impaired or Alzheimer person to identify the texts on the printed labels or books, names of the objects in real time, and names of the known person using face detection and Sign languages. To read the texts from books and printed documents an OCR (Optical Character Recognition) method is used to convert the Image to Texts. Once the images converted to text. Using Machine Learning it is ease to detect the known faces, real time objects and text to audio. The Predicted output will be Convert to Audio using Text to Speech API. The Main objective is to design a tool which supports multiple disabilities and provides solution efficiently. This Concept is designed to suit people with Dementia (Alzheimer) and for people who are visually impaired.**

Keywords*— *Automatic image extraction, Artificial Intelligence, Optical Character Recognition, Tesseract, Face detection, object detection.

I. INTRODUCTION

Brain is the primary organ of human body. The diseases that affect brain is crucial to handle, mostly once changes occur it is irreversible in severe cases. Dementia is the clinical syndrome that affect the memory skills. Alzheimer's disease is the most common cause of Dementia. Alzheimer's first appear in their middle-60's. It is estimated that more than 5.5 million people are having Alzheimer's disease. The Alzheimer's symptoms include memory loss, language problem, behaviour changes. The non-memory aspect symptoms are difficulty in word finding, vision issue and impaired reasoning. The Alzheimer's can be classified as Mild Alzheimer's disease, moderate Alzheimer's disease and severe Alzheimer's. The Cause of Alzheimer's Disease are some of genetic component for early onset Alzheimer's whereas complex series of brain change begin from Late – onset Alzheimer's. The Alzheimer's disease will first damage the hippocampus region in brain which is responsible for informing memories. Slowly it spreads to the other parts of the brain and the brain starts reducing in the size and in the last stage the brain is reduced majorly in size.

II. RELATED WORKS

[1] Ayman S. El-Baz used Personalized Computer-Aided Diagnosis for Mild Cognitive Impairment in Alzheimer's Disease based on sMRI and 11C PiB-PET Analysis The main contribution of the proposed work is to present a personalized (i.e., local/brain regional) computer-aided diagnosis (CAD) system for early diagnosis of AD from two perspectives, functional and structural to assist diagnosis using the sMRI methods. The main contribution of the proposed work is to present a personalized (i.e., local/brain

regional) computer-aided diagnosis (CAD) system for early diagnosis of AD from two perspectives, functional and structural to assist diagnosis. In other words, the proposed system uniquely yields local/regional diagnosis by combining 11C PiB positron emission tomography (11C PiB PET), which provides functional diagnosis, with structural magnetic resonance imaging (sMRI), which provides structural diagnosis. To the best of our knowledge, this is the first work to combine sMRI and the 11C PiB PET for local/regional early diagnosis of AD. [2] Guo Haibing used Resting State fMRI and Improved Deep Learning Algorithm for Earlier Detection of Alzheimer's Disease to discuss about the benefits of improved deep learning algorithms from recognizing high-dimensional information in healthcare and can lead to the early diagnosis and prevention of Alzheimer's disease. The suggested approach incorporates effectively biased neural network functionality and allows a reliable Alzheimer's disease recognition. In comparison with conventional classifiers depends on time series R-fMRI results, the proposed deep learning algorithm has improved significant and in the best cases, the standard deviation reduced by 45%, indicating the forecast model is more reliable and efficient in relation to conventional methodologies.

III. PROPOSED WORK

This proposed system idea is to develop a smart and machine learning camera-based detection system for visually impaired and Alzheimer person to identify the texts on the printed labels, or books, names of the objects in real time, and names of the known person using face detection. The Predicted output will be Convert to Audio using Text to Speech API.

In the Existing System many solutions are based on the Sensors and Devices. Since our Solution does not carry lot of equipment's like sensors and devices the cost automatically comes done which becomes very feasible and very portable and could be used anywhere, thus becoming the affordable and effective solution. [i]For family member recognition we have used Local Binary Pattern Histogram (LBPH) and Haarcascade frontal face. [ii]For real time object-detection we have used caffemodel. [iii]For Optical Character Detection (OCR) we have used Py-tesseract.

3.2.1. Family Member Recognition with Audio Feed-back system.

Passive-Aggressive algorithms are used for large-scale learning. It is one among the online-learning algorithms. In online machine learning algorithms, the input data comes in sequential order and the machine learning model is updated in sequential order, as opposed to conventional batch learning, where the entire training dataset is used at once. For family member recognition we have used Convolutional Neural Network (CNN).

3.2.2. Face Detection: Face detection is one among the fundamental applications used in face recognition technology. Facebook, Amazon, Google and other companies have different implementations of it. Before they recognize a face, their software must be able to detect it first. Amazon has developed a system for real time face detection and recognition using cameras. Facebook uses it mainly on photos that their users upload in order to suggest tagging friends. Here, application of face detection is used to some photos took using OpenCV with Python. OpenCV is an opensource software library that allows developers to access routines in API (Application Programming Interface) used for computer vision applications. The version here used was developed for Python called OpenCV-Python. To install, make sure that PIP (Python Package Index) installed with Python and run from the command line (Windows command prompt or Linux/MacOS terminal):

pip install opencv-python

This will install the main modules on the system for use with OpenCV.

Here, implementation of a system for locating faces in digital images. These are in JPEG format.Before we continue, we must differentiate between the face recognition and face detection. They are not same, but one depends on the other. In this case face recognition needs face detection for making an identification to "recognize" a face using dataset. Face detection uses classisfiers which are the algorithms that detects what is either a face (1) or not a face (0) in an image. Classifiers have been trained to detect the faces using thousands to millions of images in order to get more accuracy.

3.2.3 HaarCascades: A sequence of rescaled "square-shaped" functions which together form a wavelet family or basis. It is based on the Haar Wavelet technique to analyze pixels in the image into squares by function. This uses machine learning techniques to get a high degree of accuracy from what is called "training data". This uses "integral image" concepts to compute the "features" detected. Haar Cascades use the Adaboost learning algorithm which selects a small number of important features from a large set to give an efficient result of classifiers.Note that **face recognition** is different from **face detection**:

3.2.4 Face Detection: It has the objective of finding the faces (size and location) in an image and extract them to be used by the face recognition algorithm. **Face Recognition**: With the facial images already extracted, cropped, resized and usually converted to grayscale, the face recognition is responsible for finding characteristics which best describe the image.

This is the common difference between Face Detection and Face Recognition. Face detection is about locating and sizing, while face recognition is about identifying. Among

the other face recognition methods LBPH has more accuracy and easy to implement.

3.2.5 Convolutional Neural Network (CNN):

Convolutional Neural Networks is a powerful algorithm for image processing. These algorithms are currently the best algorithms in the automated processing of images. CNN is specialized for applications in image & video recognition. CNN is mainly used in image analysis like Image recognition, Object detection. Images contain data of RGB combination. Matplotlib is used to import an image into memory from a file.The computer doesn't see any image, all it sees is an array of numbers. Colour images are stored in 3-dimensional arrays. The first two dimensions is about the height and width of an image (the number of pixels). The last dimension of image corresponds to the red, green, and blue colours present in each pixel.

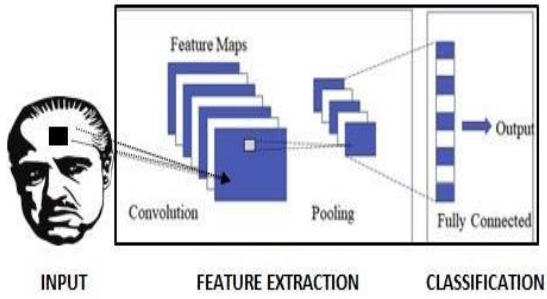


Fig. 1. CNN Layers

There are three types of layers in Convolutional Neural Networks (CNN):

Convolutional Layer, Pooling Layer and Fully-Connected layer Convolutional Layer: In a neural network each input neuron is connected to next hidden layer. In convolution layer, only a small region of input layer neurons connected to the neuron hidden layer. Pooling Layer:

```
ret, img =cam.read()
img = cv2.flip(img, 1) # Flip vertically

gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

faces = faceCascade.detectMultiScale(
    gray,
    scaleFactor = 1.2,
    minNeighbors = 5,
    minSize = (int(minW), int(minH)),
   )

for(x,y,w,h) in faces:

    cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0), 2)

    id, confidence = recognizer.predict(gray[y:y+h,x:x+w])

    # Check if confidence is less then 100 ==> "0" is perfect match
    conf=round(100 - confidence)

    if (conf > 35):
        img_text = names[id]
        confidence = "  {0}%".format(conf)

        if str(img_text) not in found :
            say(str(img_text))
            found.add(str(img_text))
```

Fig. 2. Sobel Edge Detection of Sign Language Gesture

The pooling layer is used to reduce the dimension of feature map. There are multiple activation & pooling layers inside the hidden layer of the CNN. Fully-Connected layer: Fully Connected Layers +will form the last few layers in network. The input to the fully connected layer is from the output of either final Pooling or Convolutional layer, which is flattened and then fed into the fully connected layer.

The steps involved are:

(i)Convolution Operation

(ii)Input Image

(iii)Feature Detector.

(iv)Feature map.

(v)Rectification (carrying only positive values)

(vi)Loading of dataset

(vii)Preparation of dataset

(viii)Creating a model

(ix)Evaluate the model using k-fold cross-validation

(x)Result of evaluation

(xi)Make predictions

(xii)Apply $D=\sqrt{\sum_{I=1}^{n}(\text{hist1} - \text{hist2})}$

(xiii)The algorithm output is the ID from the image with the closest histogram.

(xiv)The algorithm returns the distance value which is used for confidence value.

A. Feature Extraction

HaarCascades use machine learning techniques in which a function is trained from a lot of positive and negative images. The process in the algorithm is feature extraction. In feature extraction, the algorithm uses training data to identify features that it can consider a face.The training data used here is an XML file called: haarcascade_frontalface_default.xml.

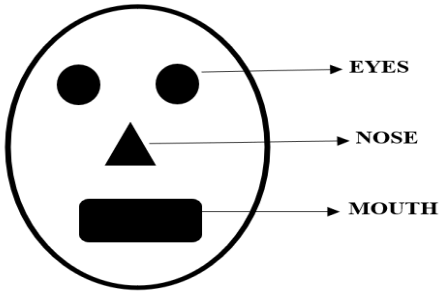


Fig. 3 Facial Features

B. Advantages

[i]Face Detection and Face Recognition Technology is straightforward to integrate, and most solutions are compatible with the bulk of security software. [ii]Helps to find missing people. [iii]Face Detection enables the automation of the identification process, which saves time and improves accuracy.

C. ARCHITECTURE DIAGRAM

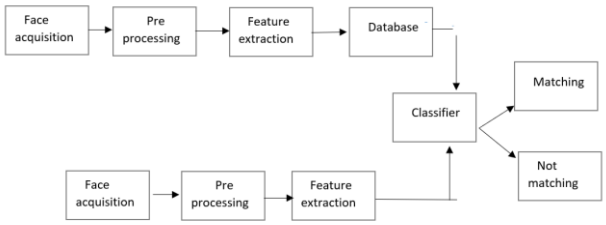


Fig. 4 Family Member Recognition with Audio Feed-back system

D. Real Time Object Recognition with Audio Feed-back system

Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Well-researched domains of object detection include face detection and Pedestrian Detection. Object detection has applications in many areas of computer vision, including image retrieval and video surveillance. Object detection is the procedure to detect the objects in real world. For example: dogs, cars, humans, birds, aeroplane etc. In this process we can detect the presence of any still object with much ease. Another great thing that can be done with it is that the detection of multiple objects in a single frame can be done easily. For real time object-detection we have use opencv caffe model along with SSD.

E. SINGLE SHOT DETECTOR (SSD)

Single Shot detector (SSD)takes only one shot to detect multiple objects present in an image using multibox. It is faster in speed and high-accuracy object detection algorithm SSD has two components: backbone model and SSD head. Backbone model usually a pre-trained image classification network as a feature extractor. The SSD head is just one or more convolutional layers added to the backbone and the outputs are interpreted as bounding boxes and classes of objects in the spacious location of the final layers activations. High detection accuracy in SSD is achieved in the usage of multiple boxes or filters with different sizes, andratio for object detectiThe SSD architecture is a single convolution network that learns to predict bounding box locations and classify these locations in one pass. Hence, SSD 31 can be trained end-to-end. The SSD network consists of base architecture (MobileNet in this case) followed by several convolution layers.

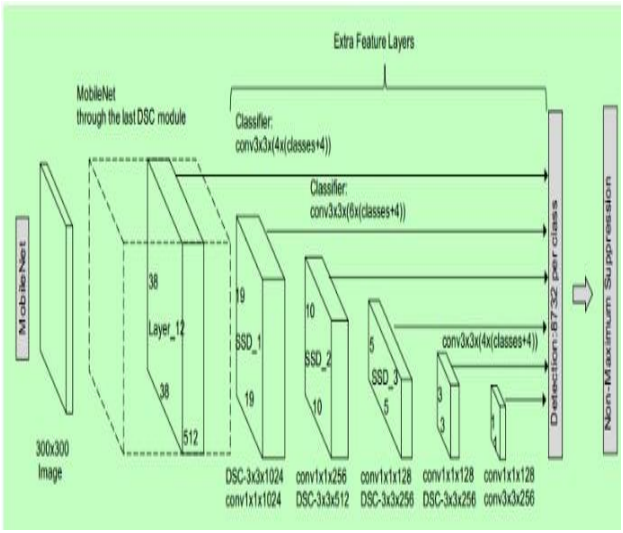


Fig. 5 Model

F. CAFFEMODEL

A CAFFEMODEL is a machine learning model created by Caffe. It contains image classification or image segmentation model that has been trained using Caffe. CAFFEMODEL files are created from PROTOTXT files. Caffe (Convolutional Architecture for Fast Feature Embedding) is the deep learning framework that allows users to create image classification and image segmentation models. Initially, users create and save the models as plain text PROTOTXT files. After user trains and refines their model using Caffe, the program saves the user's trained model as a CAFFEMODEL file. CAFFEMODEL files are the binary protocol buffer files. You cannot open, examine, and edit them in a source code editor, as you would PROTOTXT files. CAFFEMODEL files are meant to be integrated into applications that can utilize the trained image classification and image segmentation models, using Caffe.
The flow of this process is as follows: Initially, import the necessary packages numpy, imutils and others.

```
# import the necessary packages
from imutils.video import VideoStream
from imutils.video import FPS
import numpy as np
import argparse
import imutils
import time
import cv2
import os
from Models.audio import say
```

Fig. 6 Construct the argument parse and parse the arguments

```
ap = argparse.ArgumentParser()
ap.add_argument("-p","--prototxt", required=False
ap.add_argument("-m", "--model", required=False,d
ap.add_argument("-c", "--confidence", type=float,
args = vars(ap.parse_args())
```

Fig. 7 Construct the argument 2

Initialize the list of class labels MobileNet SSD was trained to detect, then generate a set of bounding box colors for each class.

```
CLASSES = ["None", "aeroplane", "bicycle", "bird", "boat",
        "bottle", "bus", "car", "cat", "chair", "cow", "diningtable",
        "dog", "horse", "motorbike", "person", "plant", "sheep",
        "sofa", "train", "monitor"]
COLORS = np.random.uniform(0, 255, size=(len(CLASSES), 3))
```

Fig . 8 Load our serialized model from disk

```
print("[INFO] loading model...")
net = cv2.dnn.readNetFromCaffe(args["prototxt"], args["model"])
```

Fig. 9 Initialize the video stream, allow the camera sensor to warmup and initialize the FPS counter.

```
print("[INFO] starting video stream...")
vs = VideoStream(src=0).start()
```

Fig. 10 If the live object is compared with the object in the classes, the output will be given in the form of audio.

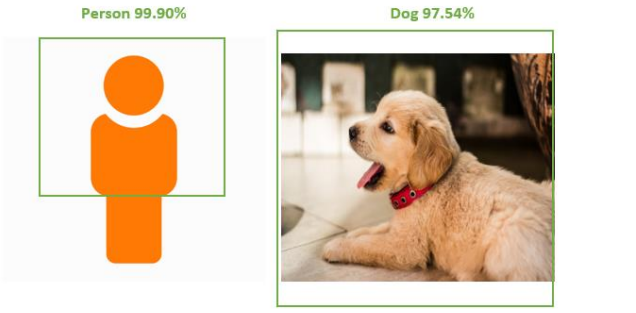


Fig. 11 Sample

[i]People counting is also a part of object detection which can be used for various purposes like finding person or a criminal. [ii]Caffe was developed as a faster and far more efficient alternative to other frameworks to perform object detection. [iii]It takes part in major fields such as face ID of Apple or the retina scan used in all the sci-fic movies [iv]For object detection, we use Caffe because it has more performance than TensorFlow by 1.2 to 5 times.

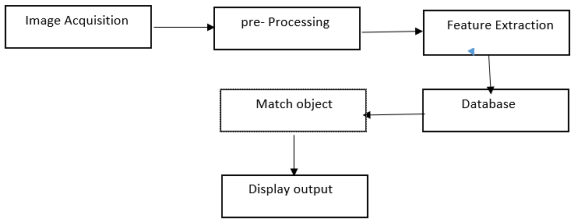


Fig. 12 Real Time Object Recognition with Audio Feed-back system

G. Optical Character Recognition with Audio Feed-back system

Python-tesseract is an optical character recognition (OCR) tool for python. That is, it will recognize and "read" the text embedded in images.Additionally, if used as a script, Python-tesseract will print the recognized text instead of writing it to a file.For Python-tesseract is a wrapper for Google's Tesseract-OCR Engine. It is also useful as a stand-alone invocation script to tesseract, as it can read all image types supported by the Pillow and leptonica imaging libraries, including jpeg, png, gif, bmp, tiff, and others. Additionally Optical Character Detection (OCR) we have used Py-tesseract.

H. Optical Character Recognition

Optical character recognition (OCR) is sometimes referred to as text recognition. An OCR program extracts and repurposes data from scanned documents, camera images and image-only pdfs. OCR software singles out letters on the image, puts them into words and then puts the words into sentences, thus enabling access to and editing of the original content. It also eliminates the need for manual data entry.

OCR systems use a combination of hardware and software to convert physical, printed documents into machine-readable text. Hardware such as an optical scanner or specialized circuit board copies or reads text; then, software typically handles the advanced processing.

OCR software can take advantage of artificial intelligence (AI) to implement more advanced methods of intelligent character recognition (ICR), like identifying languages or styles of handwriting. The process of OCR is most commonly used to turn hard copy legal or historical documents into pdf documents so that users can edit, format and search the documents as if created with a word processor.

I. Py-Tesseract

In geometry, the tesseract is the four-dimensional analogue of the cube; the tesseract is to the cube as the cube is to the square. Just as the surface of the cube consists of six square faces, the hypersurface of the tesseract consists of eight cubical cells. The tesseract is one of the six convex regular 4-polytopes.

The py-tesseract works as follows:

Initially. Initiate the camera ON to capture text image.

```
import os
tessdata_dir_config = '--tessdata-dir "C:\\Program Files (x86)\\Tesseract-OCR\\
cam = cv2.VideoCapture(0)
```

Fig. 13 Construct the argument parse and parse the arguments. Use either the live image or saved image to get output.

```
ap = argparse.ArgumentParser()
ap.add_argument("-i", "--image", required=False,default="Samples/test.png",
        help="path to input image to be OCR'd")
ap.add_argument("-p", "--preprocess", type=str, default="blur",
        help="type of preprocessing to be done")
args = vars(ap.parse_args())
```

Fig. 14 Load the example image and convert it to grayscale

```
image = cv2.imread(args["image"])
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

cv2.imshow("Image", gray)
```

Fig. 15 Check to see if we should apply thresholding to preprocess the image

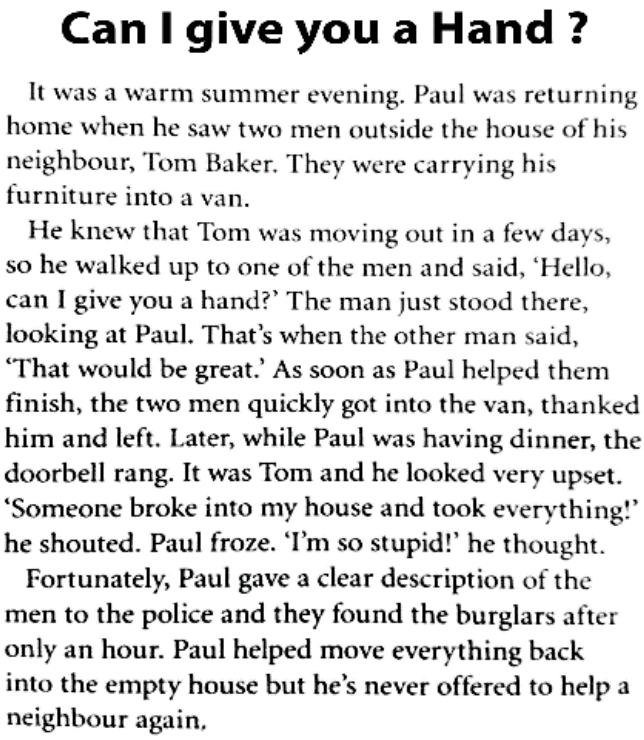
```
if args["preprocess"] == "thresh":
        gray = cv2.threshold(gray, 0, 255,
                cv2.THRESH_BINARY | cv2.THRESH_OTSU)[1]
```

Fig. 16 Make a check to see if median blurring should be done to remove noise

```
elif args["preprocess"] == "blur":
        gray = cv2.medianBlur(gray, 3)
```

Fig. 17 Median Blur

The output of this process will be:



Can I give you a Hand ?

It was a warm summer evening. Paul was returning home when he saw two men outside the house of his neighbour, Tom Baker. They were carrying his furniture into a van.

He knew that Tom was moving out in a few days, so he walked up to one of the men and said, 'Hello, can I give you a hand?' The man just stood there, looking at Paul. That's when the other man said, 'That would be great.' As soon as Paul helped them finish, the two men quickly got into the van, thanked him and left. Later, while Paul was having dinner, the doorbell rang. It was Tom and he looked very upset. 'Someone broke into my house and took everything!' he shouted. Paul froze. 'I'm so stupid!' he thought.

Fortunately, Paul gave a clear description of the men to the police and they found the burglars after only an hour. Paul helped move everything back into the empty house but he's never offered to help a neighbour again.

[i]Information of OCR can be readable with high degree of accuracy. Flatbed scanners are very accurate and may produce reasonably quality images. [ii]Processing of OCR information is fast. Large quantities of text are often input quickly. [iii]It is cheaper than paying someone amount to manually enter great deal of text data. Moreover, it takes less time to convert within the electronic form. [iv]This process is much faster as compared to the manual information.

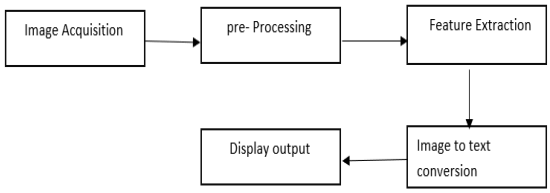


Fig. 18 Block Diagram

J. GTTS & AUDIO PLAYER: gTTS

(Google Text-to-Speech), a Python library and CLI tool to interface with Google Translate's text-to-speech API. Write spoken mp3 data to a file, a file-like object (byte string) for further audio manipulation, or stdout. Or simply pre-generate Google Translate TTS request URLs to feed to an external program. Features of gtts are Customizable speech-specific sentence tokenizer that allows for unlimited lengths of text to be read, all while keeping proper intonation, abbreviations, decimals and more; Customizable text pre-processors which can, for example, provide pronunciation corrections.

```
def cam():
        print("")
        os.system('python 06_camera.py')

def captureDataset():
        print("")
        os.system('python 01_face_dataset.py')

def faceRec():
        print("")
        os.system('python 03_face_recognition.py')

def realtime():
        print("")
        os.system('python 04_Object.py')

def OCR():
        print("")
        os.system('python 05_ocr.py')
```

Fig. 19 All these combined together to give a result

Fig. 20 This output is the combination of all the three modules which helps the dementia and visually impaired patients to overcome their hurdles.

K. FACE DETECTION

When a person wants to know their family members will go with face recognition which gives the audio of the name of the image in the dataset compared with the live image.

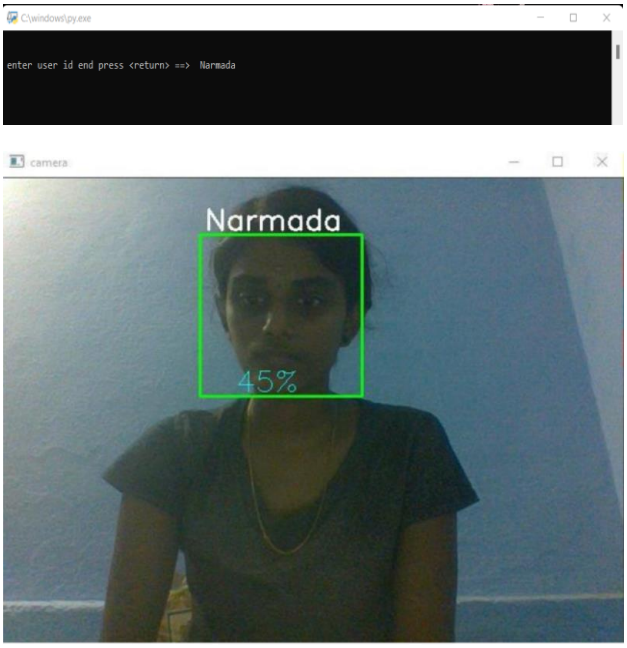


Fig. 21 Outout

L. Object Detection

When it comes to an object detection, camera is enabled showing the image in front of the camera it will name the object in audio form.

Fig. 22 Object Detection

Fig. 23 Console

M. OPTICAL CHARACTER RECOGNITION

In OCR, either the live image or saved image is given in the program it will spell it out using gtts. Here, can use the saved image and let's see the output.

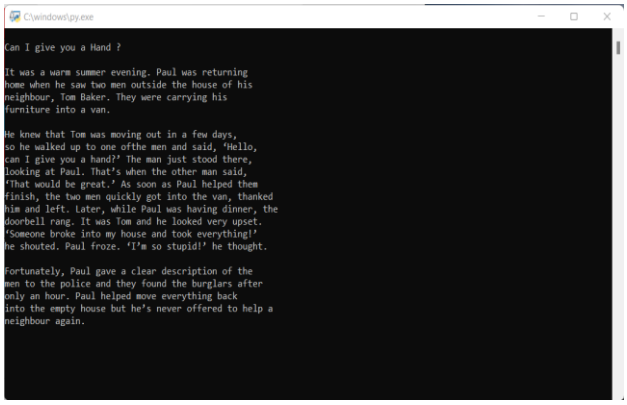


Fig. 24 Console Outout

This is the saved image since in the program it is given as saved image it will spell out the saved image. Thus, the accuracy is given by intensity of light and distance between the object and camera.

Table 1 Result

INTENSITY OF LIGHT	ACCURACY
Bright	>45%
Dull	<40%
INTENSITY OF LIGHT	**ACCURACY**
Near	>45%
Far	<40%

N. OVERVIEW OF THE PROJECT

Brain is the primary organ of the human body. The diseases that affect brain is very crucial to handle since mostly once changes occur it is irreversible in extreme cases. Dementia means the loss of cognitive functional thinking. Alzheimer's is most common cause of Dementia. Alzheimer's first appear in their mid-60's. It is estimated that more than 5.5 million people are having Alzheimer's. The Alzheimer's disease symptoms include memory loss, language problem, behaviour changes. The non-memory aspect symptoms are difficulty in word finding, vision issue, impaired reasoning and impaired judgement. The biological sign are brain images, cerebrospinal fluid and blood. The Alzheimer's disease can be classified As Mild Alzheimer's disease, moderate Alzheimer's disease and severe Alzheimer's. The Cause of Alzheimer's Disease are some genetic components for early onset Alzheimer's and Late – onset Alzheimer's begin from complex series of brain change. The other causes are genetic environment Lifestyle, Health and Detecting changes in body fluid and changes in body fluid and changes in brain can detect Alzheimer's disease. The Chemical or protein found in Alzheimer's disease are abnormal clumps (amyloid plaques), tangled

bundles of fibre (Tau Tangle) and Loss of connection between nerve cell in the brain. The symptoms of the Alzheimer's disease appear decade after it begin or onset. The deposit of protein Tau Tangle and amyloid plaques throughout brain lead to stopping of functionality of healthy neuron and once its stop function the connection, with the other neuron is lost and die. The damage will first affect hippocampus, the part of the brain essential in forming memories. Slowly it spread to other part and, the brain affected due to this chemical start shrinking and till final phase the complete brain size is shirked significantly. The Objective of the research is Provide Machine Learning Based Solution for Mildly affected Alzheimer patient Using Image Processing and Audio feedback system.

IV. CONCLUSION & FUTURE SCOPE

In this Paper, the Problem addressed of early diagnosis of Alzheimer patient by Using Machine Learning Techniques where 4 solutions provided (i) Family Member recognition Using Audio Base Feedback System which helps them identify their family members (ii) Real time object detection using audio feedback which helps them to detect the objects around them and (iii) Optical Character recognition with audio feedback system which will help them to read newspapers or books without being dependent on anyone have tested this system with Py-Tesseract, Convolution Neural Networks, LBPH, and Haarcascade and have Found Satisfactory Results which could be used to support them non-invasively.

REFERENCES

[1] Ensembles of Patch-Based Classifiers for Diagnosis of Alzheimer Diseases-Ho Yub Jung.

[2] [2] Personalized Computer-Aided Diagnosis for Mild Cognitive Impairment in Alzheimer's Disease Based on sMRI and 11C PiB-PET Analysis-Ayman S. El-Baz.

[3] [3] Resting State fMRI and Improved Deep Learning Algorithm for Earlier Detection of Alzheimer's Disease-Haibing Guo.

[4] [4] Alzheimer's Diseases Detection by Using Deep Learning Algorithms: A Mini-Review-Taha H.Rassem.

[5] [5] Machine Learning and Deep Learning Approaches for Brain Disease Diagnosis: Principles and Recent Advances- Md. Fazlul Kader.

[6] [6] Jomon Joseph and K. P. Zacharia, "Automatic Attendance Management System Using Face Recognition", International Journal of Science and Research (IJSR), 2013.

[7] [7] T. Ahonen, A. Hadid and M. Pietikäinen, "Face Recognition with Local Binary Patterns", In Proceedings of the Advances in Visual Computing, vol. 3021, pp. 469-481, 2004.

[8] [8] A. Ozdil and M. Ozbilen, "A survey on comparison of face recognition algorithms", 2014 IEEE 8th International Conference on Application of Information and Communication Technologies (AICT), pp. 1-3, 2014.

[9] [9] A. Ahmed, J. Guo, F. Ali, F. Deeba and A. Ahmed, "LBPH based improved face recognition at low resolution", 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD), pp. 144-147, 2018.

[10] [10] Ishan Ratn Pandey, Mayank Raj, Kundan Kumar Sah, Tojo Mathew and M. S. Padmini, "Face Recognition Using Machine Learning", vol. 06, April 2019.

[11] [11] Petrov, Yordan. Improving object detection by exploiting semantic relations between objects. MS thesis. Universitat Politècnica de Catalunya, 2017.

[12] [12] R. Martínez, C. Cuevas, D. Berjón, and N. García, "Detection of static moving objects using multiple nonparametric background models," pp. 4–5, 2015.

[13] [13] Maya R. Gupta, Nathaniel P. Jacobson and Eric K. Garcia, OCR binarization and image preprocessing for searching historical documents

[14] [14] How we tuned Tesseract to perform as well as a commercial OCR package, [online]Available: http://vbridge.co.uk/2012/11/05/how-we-tuned-tesseract-to-perform-as-well-as-a-commercial-ocr-package/-

[15] [15] Dirk Walther, Laurent Itti2, Maximilian Riesenhuber3, Tomaso Poggio3 and Christof Koch, Attentional Selection for Object Recognition - a Gentle Way.