*International Journal of Engineering & Technology*, 7 (4.19) (2018) 556-562

**International Journal of Engineering & Technology**

*Website: www.sciencepubco.com/index.php/IJET*

***Research paper***

# Smart Embedded Device for Object and Text Recognition through Real Time Video Using Raspberry PI

**B. Anil Kumar[1], T. Praneeth Chowdary[2*], T. Govinda Rao[3]**

*[1,3]Assistant Professor, Department of Electronics and Communication Engineering*
*GMR Institute of Technology, Rajam, India*
*[2]PG Scholar, Department of Electronics and Communication Engineering*
*GMR Institute of Technology, Rajam, India*
**Corresponding Author E-Mail: praneethchowdary143@gmail.com*

## Abstract

Object recognition, text recognition, face recognition, navigation is a challenging problem in real world scenario particularly in developing advanced technology to assist for people. The complexity of recognition for a system is difficult because of having the objects and texts are having variations in sizes, shapes, mixed with complex backgrounds and having different lighting condition.We proposed a smart embedded device that consists of five switches to recognize objects and text information from videos, images, documents and pdf files. For recognizing the object, the image data is captured by using pi camera and is processed on Raspberry pi by using SSD method for detecting objects in captured data by a single deep neural network to provide a fixed size bunch of bounding boxes and scores for the presence of object class instances in those boxes.By combining MobileNets architecture with the single shot detector framework the prediction of accuracy in detecting object is more and fast. The text information from videos are recognized by extracting a best frame using Laplacian method and performs pre-processing on the frame by applying noise removal methods. Thresholding methods are applied to improve the lucidity of the text area and Grab-cut approach is used to eliminate the unwanted backgrounds. The frame is then given to the OCR to extract the text information and was given to TTS converter to convert the text output into speech from to assist users easily.

***Keywords:*** *Object Detection; Raspberry Pi; SSD; Text extraction; Video frame extraction.*

## 1. Introduction

Upon increasing the technology there is a vast usage of using embedded devices by humans. All our lives are more contingent on embedded devices. In this digital environment these devices provide security and safety. Over 97% of processors are using in embedded systems [6]. These processors cannot be visible to users. New processors, sensors, actuators, communications and infrastructures are developing which provides a significant role in pushing the economic growth. Now a days “vision” challenges to researchers in development. This development impacts on several aspects like context-awareness, intelligence, natural interaction, restricted resources, hard real-time applications, Automotive, Medical Devices, military services, etc.

In this ongoing technology there is a large requirement of applications for text recognition, object recognition, face recognition, navigation which helps to assist people to provide them safety and security. Deep learning has become one of the leading surge for object detection. Many algorithms like YOLO, faster RCNN and SSD are developed to recognize object but SSD method provides more accuracy in recognizing objects [1]. The detecting frames in SSD framework is more than Faster RCNN.

Different datasets like PASCAL VOC800, MS COCO and GOOGLE Nets are having thousands of images but ‘Mobile Nets’ which was developed by Google researchers have millions of images with in a less storage and are designed for the applications of smart phones effectively for object detection, face attributes and large scale geo-localization [10].

For recognizing the objects in real environment this device provides magnificent results. The images are captured by using picamera. The captured data is processed by using Raspberry pi 3 model B+ board for detection and recognizing the objects [6]. The recognized objects can be visualized in monitor with object bounding and recognizing scores.

The acquisition of text data from videos are more difficult because of having the videos in different type of sources [2]. It is more difficult to extract the text because of having the text data in variant formats like scene text, scrolling text,different fonts and sizes. Moreover the complexity is more because of having huge number of frames.Different lightening conditions, focus, motion will also effect in recognition. To extract the text data from the video, the video was converted into frames and a best frame is extracted. By using image processing techniques and OpenCV framework the best frame is processed to acquire the text data [4]. The extracted text is converted into speech information to help the user in a smart way.

Several methods are available for locating the text regions in the frame. But this provides more accuracy and efficiency in acquiring the text information. Primitively the frame is pre-processed by removing noise, applying thresholding techniques and then performing segmentation on the frame [3]. The image is then processed to OCR to recognize the text information.
The text information from document and pdf files can also extracted by using this device. By using OpenCV libraries the extraction of text is more speed and accurate.By using the python language and OpenCV functions the text information in the documents files are extracted and are converted into speech form by using TTL library available by OpenCV-python.

## 2. Design

The proposed system was designed for performing five tasks torecognize objects, recognition of text from videos, images, pdf and documents in real time environment. The device consists of five switches, up on pressing the switch task assigned to them will execute. Picamera is used to capture the visual data and this information is processed by using raspberry Pi 3 model B+ board [6]. The recognized objects and text information can be visualized in monitor and through headphones. 5v 2amp power bank is connected to provide power supply for the system. The below fig.1 shows the design of the proposed system.
Raspberry Pi: Raspberry Pi is a credit card sized computer[15]. It is having weight approximately 50g. The operating voltage of Raspberry pi is 5V D.C, 2.5 amps.

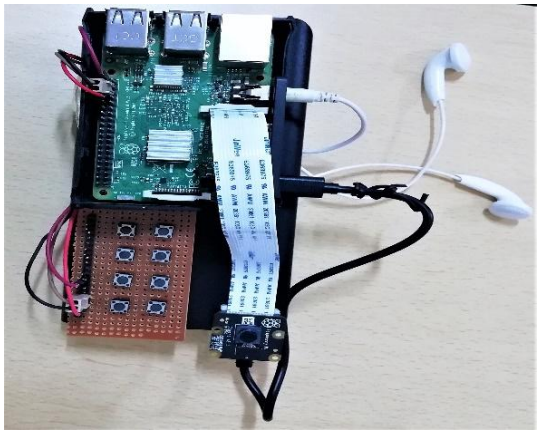


**Fig. 1:** Design of the proposed system

It is most economically efficient than a personal computer. These boards are available in different version like A, B, and advances version B+. The model B+ contains 512 MB RAM which runs on ARM II processor and the operating frequency is 900MHz [14]. The main advantage of this board is having different bootable Operating systems like Raspbian, Pidora and Raspbmc. The model B+ board can accommodate various peripherals like mouse, keyboards and Wi-Fi adapters and it is having four USB2.0 ports, Ethernet port to connect with network, GPIO to interface different sensors, switches, LED's and various other I/O devices, HDMI port to connect different peripherals like projectors, LED screens. It is having some additional features includes audio jack to connect headsets, ear phones and camera port to connect with pi camera. With these countless features Raspberry pi can be used in wide range of applications in real time.
The SD Card Slot is available to load the bootable OS and persistent for long term storage. It contains Micro USB Power Port to provide supply from source. The Raspberry Pi runs on Linux operating systems as well as there is a master version of Linux based kernel well known as "Raspbian" which can run nearly all programs which are Linux compatible. Raspbian is a free open source operating system developed for raspberry pi boards.

## 3. Implementation

### 3.1 Object Recognition

The process for detection and recognition of object in real environment can be classified into two categories: detection of object and recognition of detected object.

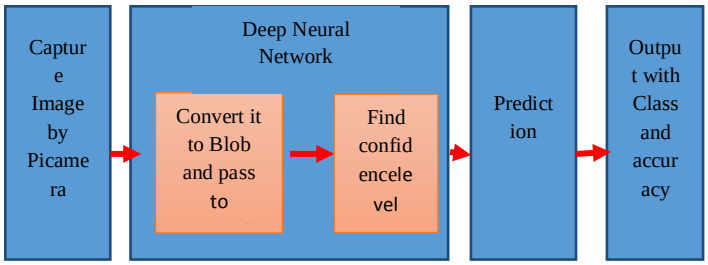



**Fig.2:** Flow diagram for detection of object in image

The proposed system initially captures the image using pi camera which is fed to raspberry pi [10].The captured image is resized to a resolution of 300×300 pixel. The image is then converted into blob by using deep neural network. By using the OpenCV the image is pre-processed using deep neural network to prepare them for classification. The blob is passed through the CNN to obtain detections and predictions of object. To detect the object the confidence level is calculated.
For evolution of object two different tasks will perform. They are classification which determines whether the object actually exists in the image are not and localization involves in determining the object location. If the object in the image is localized then the prediction takes place by comparing the object with the classes. The detected object is displayed on the screen with label of object class by a bounding box with accuracy of prediction score. The prediction of object detection is calculated by mean average precision. The below figure 2 shows the design flow of the proposed system for object recognition.This entire process is processed on raspberry pi 3. The resultant recognized object can be visualized by connecting monitor. This raspberry pi3 model B can have inbuilt VNC viewer which helps to visualize the screen in mobile phones by sharing net to raspberry pi.
The architecture in implementation of object detection and recognition was developed on Caffe framework [11].CAFFE Convolutional architecture for fast feature embedding is an opensource deep learning frame work that provides evident access to deep architectures. The library is completely written in C++ with CUDA for GPU computation and these can also be implemented in Matlab and Python languages. It was developed by Berkeley AI Research (BAIR).
Deep learning networks are structured models that are defined as a collection of inter-connected layers that work on fragments of data. It has its own method to represent nets by layer by layer. Caffe architecture mainly consists of 3 basic building blocks.

a. Data Storage
b. Layers
c. Networks

*Data Storage:* Caffe stores and communicate the data through blobs. The blobs offers accommodated memory interface for the framework to hold data. The data may be of images, parameters, and derivatives for optimization. The layer is connected next for support for computation and model. The nets are followed by connection of

more layers. Caffe provides feasible in switching conditions from CPU to GPU. The Blobs covers the computational
Processing for both CPU and GPU operation by regularizing from CPU to GPU.
*Layers:* In Caffe, layer is the core of a neural network layer. These Layers combine filters and will take the internal products, apply nonlinearities functions, loads data and calculates the losses. It takes one or more blobs as input through bottom connections and makes one or more blobs as output through top connections. The layer has primary objective in performing
Computations as setup, forward, backward.

- Setup: initialize the layer and its connections.
- Forward: takes the input from bottom and sends the output to top.
- Backward: takes the ascent with regard to the top output and computes the ascents with regard to the input and parameters.

The below figure 3 shows the blob and a simple model.

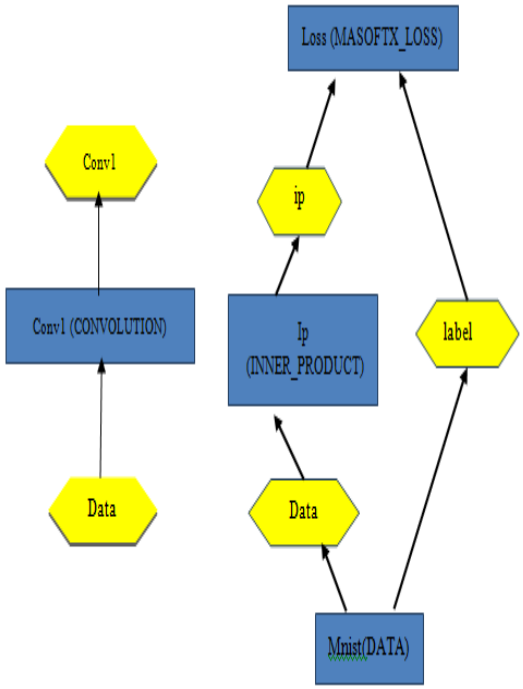



**Fig. 3:** shows the blob and sample model

*Network:* The output of every layer describes the function to perform task and every layer's backward computes the ascents from loss to learn the task. The net can be a group of layers connected in a computation graph. Caffe performs all the accounting for any of the directed acyclic graph to assure correctness of the forward and backward passes. Every net starts from a data layer and ends with loss layer in which the net loads the data from disk and the loss function appraises the goal of task.
The models and the learned models are in the form of prototxt files and binary protocol buffer files respectively. In Caffe, learning is defined by a loss function. The loss function is estimated by the forward pass of the network. Every layer get inputs from blobs and generates the output from top blobs.
The output contains loss function for every classification tasks is the softmax with loss function. The total loss in Caffe is calculated by the sum of total weighted loss over the network.Caffe trains the models fastly and uses standard stochastic gradient descent algorithm. During the training process the data layer retrieves the images from disk and transfers through multiple layersand provides the prediction layer into a classification loss layer that produces the loss and gradients. The trained network is in the form of dot prototxt file.
In our project we used MobileNet-SSD pretrained dataset which provides the researchers to use the dataset for new innovations [13].

This pretrained model uses a SSD frame work to detect the objects in the image and prediction is calculated by mean average precession [12].Before training of the neural network we must calculate all the pixel intensity of all the images in training dataset for red, green and blue channels. The three mean values are represented as Eq.1.

$$\mu_R, \mu_G, \mu_B \tag{1}$$

For larger datasets mostly pixel wise is used. Before passing of image through network we must remove the mean μ from every input channel as represented by Eq.2.

$$R = R - \mu_R, G = G - \mu_G, B = B - \mu_B \tag{2}$$

In normalization the scaling factor is included and was represented as Eq.3.

$$R = (R - \mu_R)/\sigma, G = (G - \mu_G)/\sigma \; B = (B - \mu_B)/\sigma \tag{3}$$

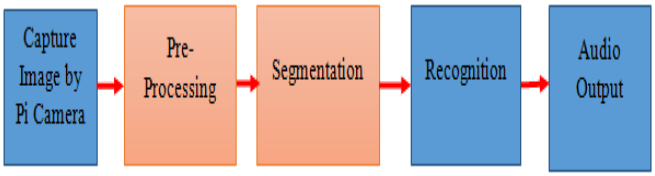



**Fig. 4:** Flow diagram for recognizing text in image

The value of variance $\sigma$ be the standard deviation across the training set. The detected objects are provided with object bounds and with accuracy.

### 3.2 Text Recognition from Images

Text recognition from the images are performed by Pre-Processing, Segmentation and Recognition.
*Noise Removal:* Image noise may be due to different sources either by sensors or from environment. So,Noise removal is mainly used to enhance the image by using erosion and dilation noise removal techniques[17]. Erosion erodes the boundaries of foreground object.
*Pre-Processing:*The main motto of the pre-processing is to make improvements in image by removing unwanted distortions and noise and to improve the perspicuity of image for further enhancements [8].Pre-processing was done by gray scale conversion, noise removal, thresholding. The above figure 4 shows the flow diagram for recognizing text from images.
*Gary Scale Conversion:* Initially, the image is converted into grayscale image.
It removes the all the pixel which are outside the kernel. It is used to remove the small white noise.Dilation is the opposite of erosion.We perform dilation of the image to increase the size or white region of the object which results from erosion by shrinking.The result of increasing of object won't let noise to come as it is already removed but helps in merging broken parts of an object**.** Using these two techniques image noise is removed.
*Thresholding:*In image processing, the functionality of thresholding is to identify the region of interest in an image by eliminating the unwanted parts in an image. Initially, Thresholding takes grayscale image and performsthe fixed or adaptive thresholding on that

image.It takes a grayscale or colour image as input and provides output as binary imagerepresenting the segmentation.

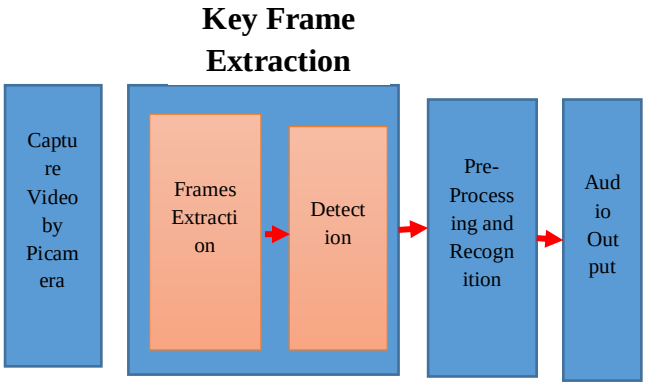



**Fig. 4:** Flow diagram for recognizing text in image

If the value lies beyond the threshold value, it is considered as background value, else it is treated as foreground value.

*Segmentation:* After applying the thresholding, the height and width of image is calculated and background of the image is identified [3]. If the Image is having the black and white background then it is considered as simple background image. For simple images the text segmentation can be performed by using line, word, or character segmentation techniques which are performed by OCR module.

If the image is having multiple background colours then it is considered as complex background images. For complex background images, the segmentation is carried by using grab-cut algorithm Grab-Cut.Grab-cut initializes background and foreground and using them it develops Gaussian Mixture Model (GMM).

Using these two distance measures, GrabCut creates the weighted graph, each node it represents the pixel. Additionally two nodes are attaches one is source node for foregroundand sink node for background pixels.The distance measures (D) to the foreground is the weight the edges from pixel to source nodes and background is the weight the edges from a pixel to the sink nodes.

Array of an image can be represented as z ($Z_1 Z_{2,\ldots} Z_n$)and Series of opacity values are represented by alpha channel $\boldsymbol{\alpha}=(\alpha_1,\ldots,\alpha_n)$ at each pixel with $0 \leq \alpha_n \leq 1$.The $\theta$ value depicts the gray-level distribution of background and foreground.

The U criticizes the data z with opacity $\boldsymbol{\alpha}$ and the K is the model vector K= ($k_{1,\ldots},k_n$) and shown in equation 4.

$$U(\alpha,k,\theta,z)=\sum_n D(\alpha_n,k_n,\theta,z_n) \tag{4}$$

Where

$$D(\alpha_{n,},k_n,\theta,z_n)=-\log \Pi(\alpha_n,k_n)+$$

$$\frac{1}{2}\log\det\sum(\alpha_n,k_n)+\frac{1}{2}[z_n-\mu(\alpha_n,k_n)]^T$$

$$\sum(\alpha_n k_n)^{-1}[z_n-\mu(\alpha_n,k_n)]$$

.

The term 'V (Smoothing term)' value results in smooth segmentation and shown in equation 5.

$$V(\alpha,z)=\gamma\sum_{(m,n)\in C}[\alpha_n\neq\alpha_m]\exp-\beta\parallel z_m-z_n\parallel^2 \tag{5}$$

$$\beta=\left(2\left\langle(z_m-z_n)^2\right\rangle\right)^{-1}$$

GrabCut algorithm finally calculates the minimum cut from the created graph and finds the minimum-cost segmentation (E) value and re-assigns these pixel values. This process continues till the convergence by relearning and constructing new graph using GMM.

$$E(\alpha,k,\theta,z)=U(\alpha,k,\theta,z)+V(\alpha,z) \tag{6}$$

The image after applying the grab-cut algorithm is shown in figure 5.

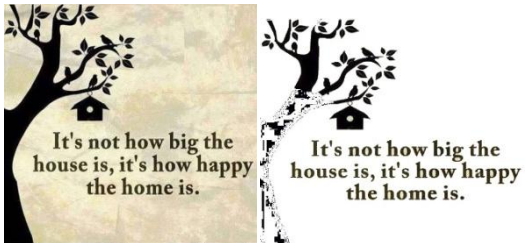



**Fig. 5:** (a) Input Image (b) after applying GrabCut

The segmented output is given for recognition to extract the text information.

*Recognition:* The Optical Character Recognition (OCR) is software tool used for automatically recognizing the characters present on the image. The recognized text information is given to TTS converter to convert into speech form.

### 3.3 Text Recognition From videos

Initially the video is captured by using pi camera. The captured video is spilt into sub videos each one having 1 sec duration by using raspberry pi [8]. Video splitting is performed by ffmpeg tools of moviepy frame work. By using ffmpeg tools input video is cropped into sub videos. From each video key frame is extracted.

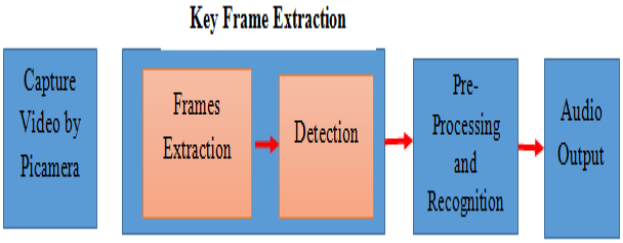



**Fig. 6:** Flow diagram for recognizing text in videos

The below fig 6 shows the flow diagram of recognizing text from videos.

Key Frame Extraction is extracting the best frame, to be analysed, from a definite number of frames [9]. Where a video is divided into multiple frames and the best out of them was selected. We take threshold value of every image based on the clarity, colour differentiation and exposure values. Based upon these threshold values the key frame can be determined.

For each frame that is obtained from key frame extraction, we find the best frame using blur detection method. Laplacian method is used to find the blur value. In Laplacian method we take grayscale and convolve them with the pre-determined Laplacian kernel. Then we find the variance from squared standard deviation. The image is treated as blurry, if the achieved variance is par below as threshold. The image is treated as not blurry, if the achieved result more the threshold.

After finding the best frame, the resulted image is performed by pre-processed, segmentation and recognition assame way for text extraction from image explained in module 2. Rotation of the image is performed by using Hough transform. Any image with different angles comes to its normal way by using Hough transform.

### 3.4 Text Recognition from Document file

Word files are binary files and are more complex files rather than plain text files [16]. The information to extract from this is more

complex because of having the information in different types of fonts, various colours and different layouts. In order to change the information in reading or writing in word file we need permissions initially. Python modules are available to make easy for opening Word documents. This makes the interaction easier.
Python has the library python-docx, by using this we can modify and create word documents easily with the extension .docx. We can get this library by installing through preferred installer program. The .docx files contains more structure when compared to plain text. Python-Docx represents this structure in different data types. The top-level or highest level represents the Document object.The below figure7 shows the flow diagram of text recognition from word and pdf files. In Word document, the text is more than the string object, it has size, color, font and some other information. Each Style object in the word document has these all attributes as collection. A Run object has same style with contiguous run of the text. Whenever the style changes, then new Run object takes place. More run objects are run simultaneously in extracting the text information.

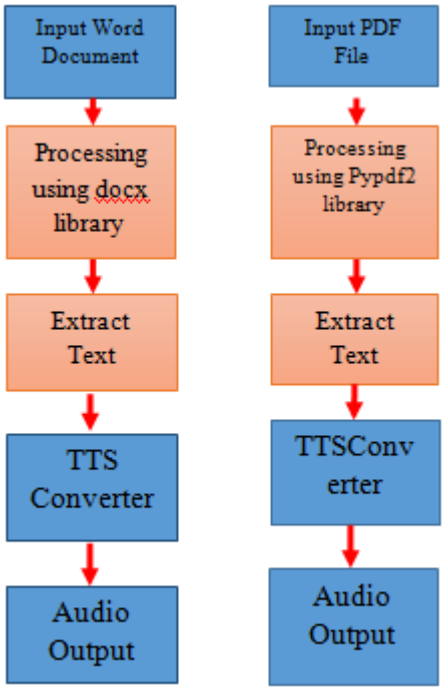



**Fig. 7:** Flow diagram for recognizing text information from documents and pdf files

### 3.5 Text Recognition from Pdf File

Portable Document Files (PDF) uses .pdf extension [16]. PyPdf2 module is used to extract the text from pdf files. Pdf files layout the text in easy manner, it is easy for the people to read and print. And it is not straight forward to parse as plain text. Pypdf2 is unable to extract the charts, images or any other media form pdf files but it is able to extract the entire test as string format.
Encryption was one of the best feature that a pdf documents have. These provides security by providing a password. The encrypted pdf can be decrypted by using existing library. Some pdf pages may be in different angles. This type of pages can also extract by using pypdf2by rotating 90 degree angles.

## 4. Experimental Results

This embedded device is tested on various images captured in real time. The images and videos are captured by Pi Camera having resolution ranges from 640*480 to 1920*1080.

### 4.1 Recognizing Objects from Images

By combining the SSD with MobileNets the detection of objects is speed and accuracy is more in detected objects. These proposed system can be highly used for detecting objects for low resolution Images.

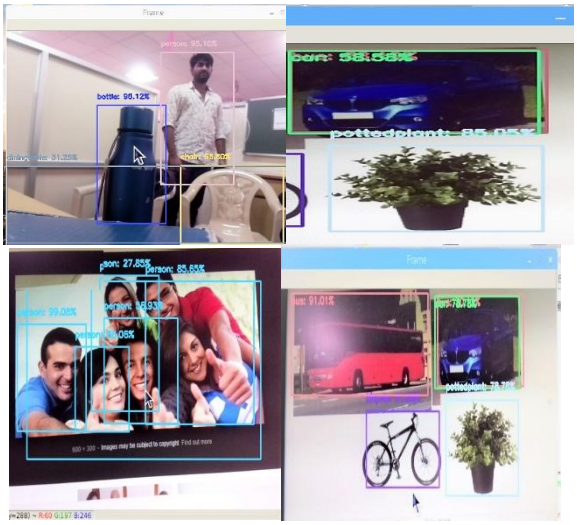


**Fig. 8:** Recognized objects in image with detecting accuracy

### 4.2 Recognizing Text from Images

Our proposed system was tested on various complex images. The recognition of text information is fast and provides better accuracy in dealing with complex backgrounds. Figure 9 shows some sample images and with detected outputs.

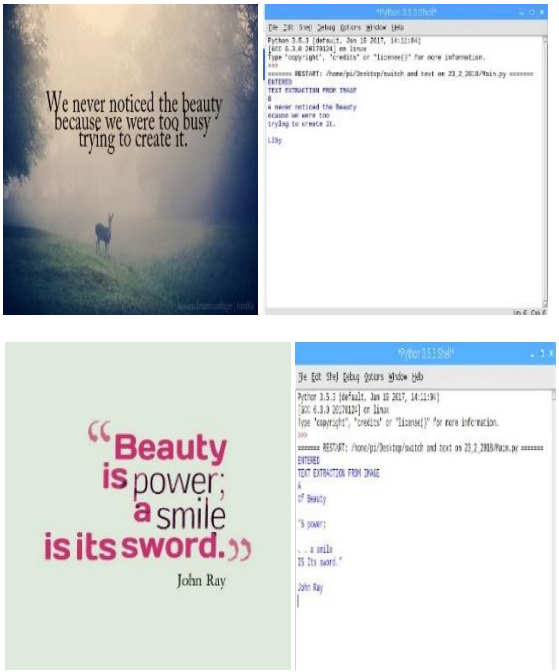



**Fig 9:** Recognized text from sample images

### 4.3 Recognizing Text from Video Frame

The Keyframe is extracted from a video and is text information is extracted. Figure 10 shows the detected keyframe as not blurry and the text information is extracted.

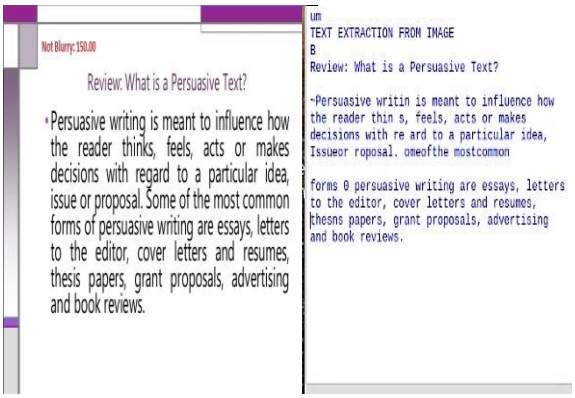
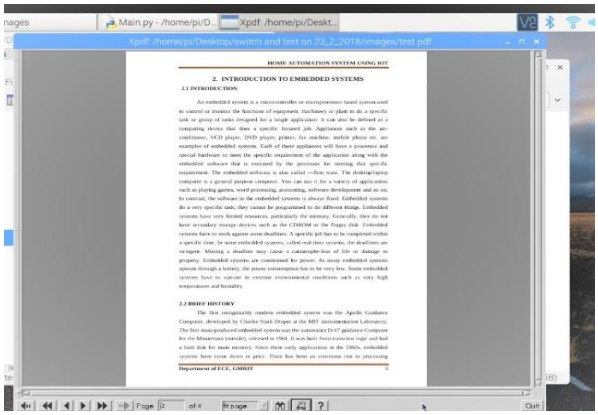


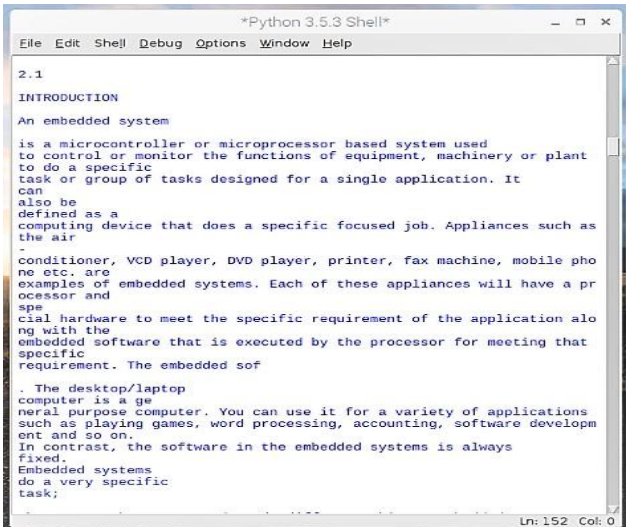

**Fig 10:** Recognized text from a key frame in video

### 4.4 Recognizing Text from Pdf File

From the pdf file the text information is extracted using python-OpenCV libraries. Figure 11 shows the recognized text from an input pdf file.



**(a)**

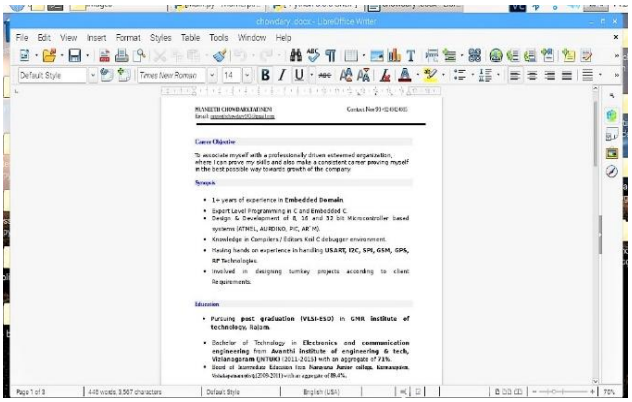



**(b)**

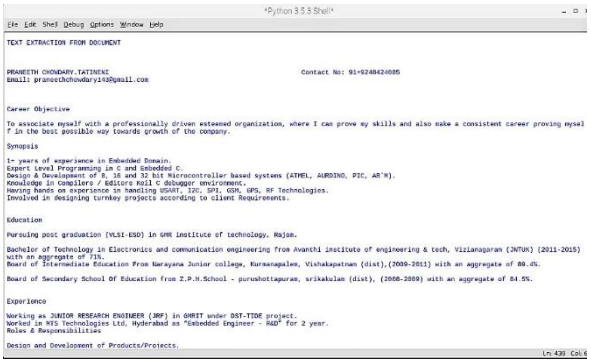
**Fig. 11:** (a) input pdf file (b) output

### 4.5 Recognizing Text from Document File

From the word document the text information is extracted using python-OpenCV libraries. Figure 12 shows the recognized text from an input word document.



**(a)**




**(b)**

**Fig. 12:** (a) input document file (b) output

## 5. Conclusion

We proposed a smart device for recognizing the objects and texts. This device assists the people for recognising the objects and extracts the text information from videos, pdf and documents in real time. Switches are provided to execute the tasks individually. For recognizing the objects, when the switch was pressed the image data is captured by using pi camera and is processed by raspberry pi 3 model B+ board. The recognized objects can be visualized in monitor with object bounds and detecting accuracy. SSD method is used for detecting objects in visual data by a single deep neural network. MobileNets architecture was used to reduce the size of the trained datasets. This device works for detecting various objects classes includes "car, chair, cow, dining table, motorbike, person, potted plant, sheep, bird, boat, bottle, airplane, bicycle, bus, cat, dog, train, sofa, TV monitor and horse" in real time.

To recognize the text information from videos. A key frame is extracted from the video by using Laplacian operator and performs pre-processing methods by applying noise removal methods and then thresholding methods are used to improve the lucidity of the text area. By using Contours and Edge Detection the text information is detected and the text regions are separated from complex backgrounds. Optical Character Recognition is used for recognition of text from the identified text patterns. The recognise text is converted into speech information using TTS module.

For recognizing the text information from documents and files the corresponding files are given to the raspberry pi board. By using the OpenCV libraries the text information is extracted and converted

into speech information. The results point out that it will be possible to acquire effective performance using features of OpenCV and Python. The effectiveness of this device increases by providing high speed processor and high resolution cameras. This work can be enhanced for moving objects and scrolling videos in future.

## References


[1] S. Ren, K. He, R. Girshick, J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39, pp. 1137-1149.

[2] Nidhin Raju, Dr. Anita H.B, "Text Extraction from Video Images", International Journal of Applied Engineering Research,2017, 12, pp.14750-14754.

[3] Santosh, L.M. Jenila Livingston. Text Detection from Documented Image Using Image Segmentation. International Journal of Technology Enhancements and Emerging Engineering Research, 2013, 1,ISSN 2347-4289.

[4] Q. Ye, D. Doermann. Text Detection and Recognition in Imagery: A Survey. IEEE Transactions on Pattern Analysis& Machine Intelligence, 2015, 37, pp. 1480-1500.

[5] Yi-Feng Pan, Xinwen Hou, Cheng-Lin Liu. A Hybrid Approach to Detect and Localize Texts in Natural Scene Images. IEEE Transactions on Image Processing, 2011, 20,pp. 800-813.

[6] S. Goyal, P. Desai, V. Swaminathan. Multi-Level Security Embedded With Surveillance System. IEEE Sensors Journal,2017, 17, pp. 7497-7501.

[7] J. Ohya,A. Shio, S. Akamatsu.Recognizing characters in scene images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1994, 16,pp. 214–220.

[8] Keechul Junga, Kwang In Kimb, K. Anil Jainc.Text information extraction in images and video: a survey. Pattern Recognition Society, 2004, 37, pp. 977-997.

[9] Divya patel. A Review Paper on Object Detection for Improve the Classification Accuracy and Robustness using different Techniques. International Journal of Computer Applications, 2015, 112, pp. 0975-8887.

[10] Adrian Rosebrock. Raspberry Pi: Deep learning object detection with OpenCV, 2017. https://www.pyimagesearch.com/2017/10/16/raspberry-pi-deep-learning-object-detection-with-opencv.

[11] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, Trevor Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. IEEE Conference on Computer Vision and Pattern Recognition, Jun 2014.

[12] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg. SSD: Single Shot Multi Box Detector. IEEE Conference onComputer Vision and Pattern Recognition,Dec 2015.

[13] Andrew G. Howard, Menglong Zhu, BoChen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. IEEE Conference on Computer Vision and Pattern Recognition,Apr 2017.

[14] B. Anilkumar, KRJ Srikanth. Design and Development of Real Time paper Currency Recognition System of Demonetization New Indian Notes by Using Raspberry Pi for visually challenged. International Journal of Mechanical Engineering and Technology, 2018, 9, pp. 884–891.

[15] Virginia Menezes, Vamsikrishna Patchava, M. Surya Deekshith Gupta.Human detector and counter using raspberry Pi microcontroller. Innovations in Power and Advanced Computing Technologies, Jan 2018.

[16] Takasshi hirano, Yuichi Okano, Yasuhiro okada, Fumio yoda. Text and Layout Information Extraction from Document Files of Various Formats Based on the Analysis of Page Description Language. Ninth International Conference on Document Analysis and Recognition,Sep2007.

[17] B. Anil kumar. Hardware Implementation of Image Processing Concepts for Mechatronics: A Survey. International Journal of Mechanical Engineering and Technology, 2018, 9, pp.876-883.

[18] S. Syed Ameer Abbas, M. Anitha, X. Vinitha Jaini. Realization of multiple human head detection and direction movement using Raspberry Pi. International Conference on Wireless Communications, Signal Processing and Networking, Mar 2017.

[19] Samruddhi Deshpande, RevatiShriram. Real Time Text Detection and Recognition on Hand Held Objects to Assist Blind People. International Conference on Automatic Control and Dynamic Optimization Techniques, Sep 2016.