Smart Delivery Box based on TensorFlow Object Detection and Recognition API

Dr.B.Muthusenthil
Associate Professor, CSE
SRM Valliammai Engineering College
Chennai, Tamil Nadu, India
muthusenthilb.cse@valliammai.co.in

Dr.V.Dhanakoti
Associate Professor, CSE
SRM Valliammai Engineering College
Chennai, Tamil Nadu, India
dhanakotiv.cse@valliammai.co.in

Dr.M.Mayuranathan
Associate Professor, CSE
SRM Valliammai Engineering College
Chennai, Tamil Nadu, India
manimayur@gmail.com

T.S.Ramachandran
Computer Science and Engineering
SRM Valliammai Engineering College
Chennai, Tamil Nadu, India
tsramachandran770@gmail.com

***Abstract*— The Object Detection Smart Delivery box operates on raspberry pi with two cameras, one camera that takes a picture of the delivery person and sends it to the user so that the user can remotely open the delivery box with the help of an android application. Another camera which is placed inside the box scans the package, recognizes it with help of "TensorFlow object detection API" and mails the picture of the package to the user. Receiver module displays the name of the package and tells what package is delivered. The user can also use Google assistant voice command to open or close the Object Detection Smart Delivery box.**

Keywords— TensorFlow; Python; Raspberry Pi; Android Application; Firebase; Arduino

I. INTRODUCTION

Object detection is the process of detecting and recognizing visual objects. It is the basis of several computer vision tasks like image captioning, object tracking etc. Object detection algorithms were developed for the first time about 20 years back when there was a lack of image representation. In 2001, Paula Viola built an object detection framework that can recognize real-life objects. It scans the images throughout the given frame in every possible area and scales to check whether any section matches a human face. There has been significant growth in E-commerce in the past years. It was expected that e-commerce sales would reach up to 4.2 trillion dollars by 2020. But the number has been growing exponentially since then. This explains the need for smarter technologies in delivering online packages to make them trouble-free for the users. [1] This paper explains about the use of Object detection and recognition in traffic signals. The TensorFlow API has been trained with the help of photographs in various traffic and climatic conditions by choosing specific deep learning models and results were analyzed. [2] Optical mark reader reorganization problems and the difficulty in virtual operating have increased human labor. The TensorFlow module has been proposed to build the model architecture on the basis of convolution neural networks. Deep learning tools in TensorFlow have been used to construct the neural network model to upgrade OMR recognition technology.

For preprocessing OMR input data and training the model to reach the requisite OMR recognition level in paper images, TensorFlow has been used. The final graphs of the TensorFlow model are presented with the help of a tool called TensorBoard. [6] The implementation of OpenCV in an embedded system like Raspberry Pi to develop a mini-standalone station for calculating people is being focused on in this paper. The main characteristic of AU-PICC (Assumption University's raspberry Pi Customer Counter) is to calculate the number of people on the chosen product along with basic face identification to avoid duplicates. The end results state that this raspberry Pi-based system can be used as a fundamental people counting station.

The outline of the paper is organized as follows: Part II is the Literature review, Part III is the System Overview, Part IV is the Result and analysis section, Part V is the Future scope section.

II. LITERATURE REVIEW

Fatma Salih; Mysoon S.A. Omer [4] enlist Raspberry pi as a Video Server. The Use of Raspberry pi microcomputers is expanding swiftly in numerous applications and projects. One of which is to use it as a video server. Raspberry Pi programming has been used to design an embedded LAN live real-time video/audio stream server. Raspberry pi camera module port captures the video and compresses it to send it using a special standard that applies HTTP so that it can be received from the network. RTP protocol is used to send the audio after capturing it through the microphone. The Raspberry pi is capable of performing a video stream of 800 × 400 at 24 frames per second.

Ying Li; Junxin Cheng; Xin Wang [5] explains An Optophone Based on Raspberry Pi and Android Wireless Communication. An optophone is developed with an aim of making sight-disabled people have the same reading perspective as that of common people. The CSI camera has been connected to the Raspberry Pi to achieve the image acquisition of physical paper reading materials. The Raspberry Pi audio interface is connected to the playback device to facilitate users with reading. The Android app communicates with the Raspberry Pi to control the GPIO.

Md. Tobibul Islam; Mohiuddin Ahmad; Akash Shingha Bappy [8] enlist Real-Time Family Member Recognition Using Raspberry Pi for Visually Impaired People. The development and estimation of the raspberry pi-based smart glass system to help blind people to recognize their family members using image processing is discussed in this paper. It delivers the name of the family member as audio information to the blind person. Raspberry Pi is one of the powerful processors in which machine learning techniques like SVM, TensorFlow can be easily integrated. This system can recognize the person in front of the blind man and provides audio name information. It has also used raspberry pi face recognition and face database creation for the identification of the family member. This device can be a great advantageous device for the blind person. This smart glass system has additional features such as low weight, extremely economical, and highly efficient.

III. SYSTEM OVERVIEW

3.1 System architecture

In this paper, we explore the Object Detection Smart Delivery Box. The delivery boy delivers products such as Newspaper, Food Package, Online shopping package, Milk etc. The camera fixed above the delivery box captures the photograph of the delivery boy and mails it to the user's phone. The user can remotely open the delivery box with the help of an android application. Another Pi-camera, which is fixed inside the delivery box scans the package, recognizes it with help of "TensorFlow object detection API" and mails the picture of the package to the user. Receiver module, which consists of Arduino UNO which is a microcontroller, XBEE receiver module that gets data from the Raspberry Pi through radio waves, speaker that tells the user what package is delivered, 16X2 LCD Display module displays the details of the package. The user can also use Google assistant voice command to open or close the Smart Delivery box. The android application consists of two activities, the first activity consists of a User Login screen which is integrated with Firebase authenticator. The second activity consists of two buttons, one to open the door and another to close the door. The top of the activity screen displays the current status of the Smart box door and the bottom of the screen displays the log files.

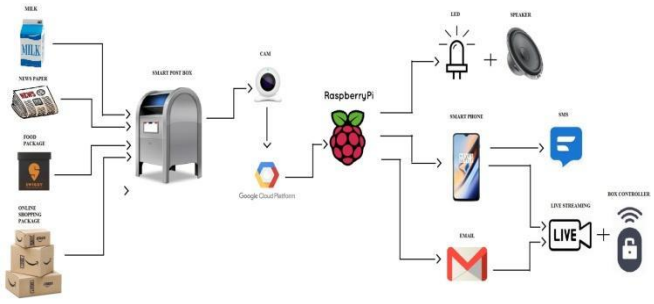

Fig. 1 Proposed system of Object Detection Smart delivery Box

Object Detection Smart Delivery Box is classified into the Following:

- TensorFlow Object Detection API: TensorFlow is developed for deep learning. ssd_mobilenet_v1_coco architecture runs on Raspberry Pi 4 to perform object detection on live video feeds from a Pi-camera or USB webcam and it is used to train neural networks to identify specific objects.

- Automated Python script: This script will start then the package is placed inside the box. It takes the picture of the package with "fswebcam" and sends it to TensorFlow API for product recognition. If the package in the box is recognized by TensorFlow, it sends a mail to the user with the name of the package.

- Android application: It consists of two activities namely the first activity consists of User Login screen which is integrated with Firebase authenticator. The second activity consists of two buttons, one to open the door and another to close the door. The top of the screen displays the current status of the Smart box door and the bottom of the screen displays the log files.

- Receiver module: It consists of Arduino UNO which is a microcontroller, XBEE receiver module that gets data from the Raspberry Pi through radio waves, speaker that tells the user what package is delivered, and 16X2 LCD Display module displays the details of the package.

- Result: Before this application, the user needs to be available in their homes for successful deliveries. This product makes it easy for both the user and the retailer as the user can collect their package from anywhere safely without the problem of theft or wrong deliveries.

3.2 TensorFlow Object Detection API

TensorFlow is an end-to-end open-source software library for machine learning and deep learning. TensorFlow object detection API can be installed on low performance ARM based single board computers and can also run on Smartphones which makes it a simple and flexible architecture to develop and execute new ideas from concept to code. The following steps are used to install TensorFlow on Raspberry pi which is a single board computer.

- Install a Rasbian OS onto a SD card with the help of the official Raspberry pi imager. After completion, eject the SD card and insert in the raspberry pi. Connect the monitor, keyboard, mouse, LAN cable for Internet.

- Once the raspberry pi is booted, update the raspberry pi with the following command " sudo apt-get update && apt-get dist-upgrade" Wait until the raspberry pi updates and reboots.
- Install Dependency packages for TensorFlow using command " sudo apt-get install libatlas-base-dev libopenblas-dev libblas-dev"
- Remove the old version of TensorFlow if installed. Download TensorFlow.whl from the official website.
- Install Tensorflow.whl with the following command " sudo -H pip3 install tensorflow.whl && reboot"
- After raspberry pi reboots, open terminal and navigate to the TensorFlow directory and execute the tensorflow.sh file.

Fig. 2 TensorFlow 2.0 on Raspberry pi with Pi-cam

3.3 Automated Python script

Python is one of the popular and demanding programming and scripting languages because it is suitable for developing very simple code to very complex IT applications. Python is one of the apt languages for controlling the GPIO pins in the Raspberry pi and also supports I2C communication. It is useful for recording logs and errors which make it easy for debugging.

- In the beginning of the python script, initialize the GPIO pins for the flashlight which is controlled by a five volt mechanical relay module.
- The script checks whether the button is pressed by the delivery person. Once the button is pressed, the script takes the picture of the person with the following command " sudo fswebcam –r 640 X 480 human.jpg"
- The script mails the human.jpg file as attachment to the predefined user E-mail address by the following command " server.send_mail('myemail@gmail.com', mail)"
- The script updates the current status of the button to the firebase relational database with the command " db.child("box").update({"button": "on/off" })"
- When the package is placed inside the box, the picture is taken by pi-camera and sent to Tensorflow.sh for recognition of the package.

```
while True:
    GPIO.setmode(GPIO.BCM)
    GPIO.setup(20, GPIO.IN, pull_up_down=GPIO.PUD_UP)
    input_state=GPIO.input(20)

    if input_state == False:
            print('button on')
            print('take a pic')
            os.system('sudo fswebcam -r 640x480 human.jpg')
            print('sending mail')
            server.send_mail('kaillinux770@gmail.com', mail)
            db.child("box").update({"button":"on"})
            time.sleep(5)
            db.child("box").update({"button":"off"})
            i=100
            while(i>0):
                  print("waiting for owner to open the box "+str(i)+"left")
                  value1 =db.child("box").child("door").get()
                  doorinfb1=value1.val()
                  if doorinfb1=='open' :
                        print("opening door")
                        relay(23)
                        db.child("box").update({"door":"close"})
                        print("door closed")
                        time.sleep(5)
                        #on light
                        GPIO.setmode(GPIO.BCM)
                        GPIO.setup(22, GPIO.OUT)
                        GPIO.output(22,GPIO.LOW)
                        #classifiaction
                        os.system('python3 smartbox.py')
                        #get the type of object
                        file=open("test.txt","r")
                        x=file.read()
                        print (x)
                        #send to xbee
                        xbee(x)
                        #off light
                        GPIO.cleanup()
                        i=0
                  i-=1
    else :
            print('button off')
```

Fig. 3 Code for Automated Python script

3.4 Android Application

Android application consists of activities which performs onCreate(), onstart(), onresume(), onrestart(), onpause(), onstop() in every instance of the application process.

- In the setEvenListener() function that displays the function log of the door which is closed or not added the time and date of the incident using SimpleDateFormat.
- In function setonClickListener() is used to wait for the user interaction to press the button or to hold the button. Once the button is pressed, the OnClick() function executes the block of code.
- The data in the firebase relational database can modify, create, delete using the following command "mRootRef.child("box").child ("door").setvalue("close")"

```
    private void setEventListener() {
        final MediaPlayer closed =MediaPlayer.create(this,R.raw.closed);
        final MediaPlayer open =MediaPlayer.create(this,R.raw.open);
        mButtonRemove.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                SimpleDateFormat formatter =new SimpleDateFormat("dd/MM/yyyy HH:mm:ss");
                Date date =new Date();

                try{
                    closed.start();
                    mRootRef.child("box").child("door").setValue("close");
                    mTextview2.append("Door closed  at  "  + formatter.format(date)+ "\n");
                }catch(Exception e){
                    e.printStackTrace();
                }
            }
        });
        mButtonPush.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                SimpleDateFormat formatter1 =new SimpleDateFormat("dd/MM/yyyy HH:mm:ss");
                Date date1 =new Date();
                try{
                    open.start();
                    mRootRef.child("box").child("door").setValue("open");
                    mTextview2.append("Door opened at  "  + formatter1.format(date1)  + "\n");
                }catch(Exception e){
                    e.printStackTrace();
                }
            }
        });
    }
}
```

Fig. 4 Code for Android Application

- The following flowchart gives an overview of the all possible use cases in the object detection smart delivery box.
- The delivery person presses the button and the notification is sent to the user. In this case, the user can grant access to the delivery person by opening the door or denying it.
- The TensorFlow can only find the products that are trained earlier with the ssd_mobilenet_v1_coco module. In this case, products that are not recognized are not classified in the given TensorFlow table. These product images are sent as mail with the name as undefined.

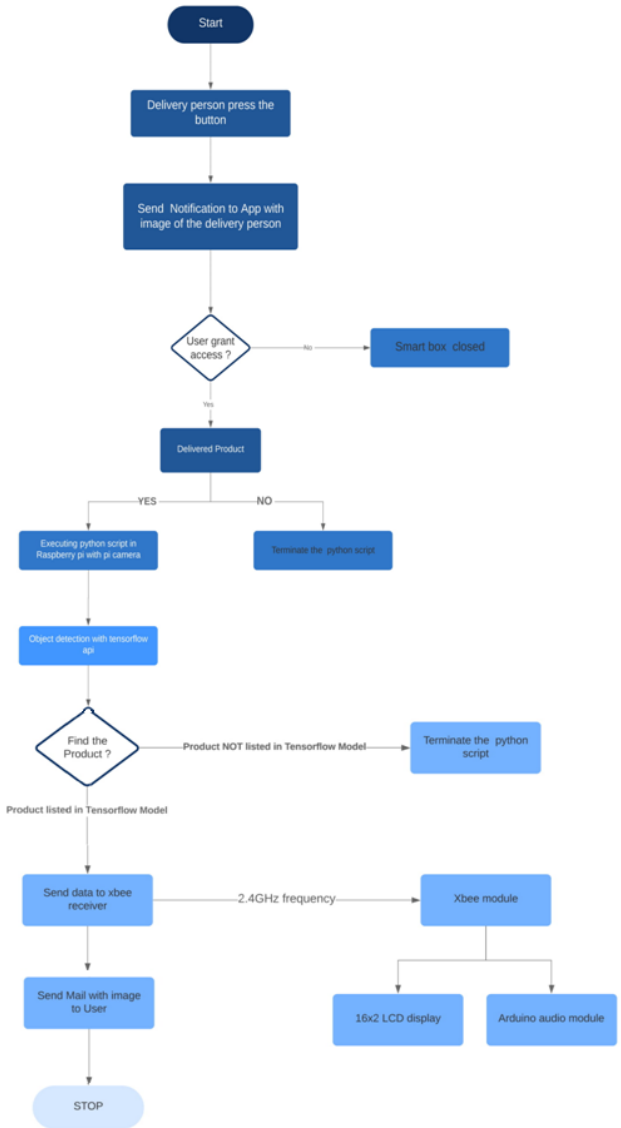

Fig. 5 Flowchart for Smart post box

3.5 Receiver model

Receiver module, which consists of Arduino UNO which is a microcontroller, XBEE receiver module that gets data from the Raspberry Pi through radio waves, speaker that tells the user what package is delivered, 16X2 LCD Display module displays the details of the package.

- The Arduino UNO is connected to the XBEE receiver module through the I2C communication protocol. It sends and receives data packages from raspberry pi with the help of 2.4g radio waves.
- The SD card module communicates with Arduino via data pins “MISO, MOSI, SCK, and CS”. The SD card stores small WAV files which consist of text to speech preloaded voice.
- The 16X2 LCD display module shows the current status of the object detection smart delivery box and indicates the user by displaying the name of the product in the box.

Fig. 6 Receiver model

IV. RESULT & ANALYSIS

Before this product came into practice, it was difficult for the delivery person to deliver the packages without the presence of the user. Because they need to ensure that the package is safe and delivered to the right person. The problem is that when the delivery person leaves the package at the user’s doorstep, it is prone to theft or damage. This Object detection smart delivery box ensures the remote collection of packages by the user without any hassle.

- When comparing ssd_mobilenet_v1_coco and ssd_inception_v2_coco models both have high speed processing, coco mAP values are 21, 24 respectively. The ssd_mobilenet_v1_coco is a good choice for low performance ARM based single board computers.
- The raspberry pi camera has a low frame rate when compared to the USB camera. So, the Pi-cam is used to take the picture of the product and send it to TensorFlow object detection API. The USB camera is used to take a clear picture of the delivery person and send the mail to the user.

Fig. 6 Object detection smart delivery box

- In the below fig.8, the current status of the door in the object detection smart delivery box is displayed at the top of the activity page. This status is retrieved from firebase relational database.
- The black screen represents the log of the door with the date and time of the event. This is useful for easy debugging in the program.
- At the bottom of the screen, open door button and close door button are used to send a modified request to the entity in the firebase relational database.

Fig. 7 Android application login activity

Fig. 8 Android application main activity

V. FUTURE SCOPE

This product has an added feature of sanitizing the packages with the help of UV light which ensures that the package is free from any kind of germs. This acts as an added advantage to the users from communicating any infections through the packages thereby making their online shopping experience stress-free. It also has a feature of voice interaction that enables the user to speak with the delivery person from a remote position.

VI. CONCLUSION

Although online shopping has made our life easier to purchase any products in a click, the time of delivery of these packages has always been a worry factor for the users. This Object detection smart delivery box makes it easier for the user to receive any kind of delivery packages from anywhere by making the entire process contactless and automated at the same time ensuring the safety of the packages.

REFERENCES

[1] Irfan Kilic; Galip Aydin — Traffic Sign Detection And Recognition Using TensorFlow' s Object Detection API With A New Benchmark Dataset
DOI: 10.1109/ICEE49691.2020.9249914
Available: https://ieeexplore.ieee.org/document/9249914

[2] Yunxia Ju; Xichang Wang; Xiangxi Chen — Research on OMR Recognition Based on Convolutional Neural Network TensorFlow Platform
DOI: 10.1109/ICMTMA.2019.00157
Available: https://ieeexplore.ieee.org/document/8858693

[3] Cheng-Hsiung Hsieh; Dung-Ching Lin; Cheng-Jia Wang; Zong-Ting Chen; Jiun-Jian Liaw — Real-Time Car Detection and Driving Safety Alarm System With Google TensorFlow Object Detection API
DOI: 10.1109/ICMLC48188.2019.8949265
Available: https://ieeexplore.ieee.org/document/8949265

[4] Fatma Salih; Mysoon S.A. Omer — Raspberry pi as a Video Server
DOI: 10.1109/ICCCEEE.2018.8515817
Available: https://ieeexplore.ieee.org/document/8515817

[5] Ying Li; Junxin Cheng; Xin Wang — An Optophone Based on Raspberry Pi and Android Wireless Communication
DOI: 10.1109/AEECA49918.2020.9213587
Available: https://ieeexplore.ieee.org/document/9213587

[6] Tussanai Parthornratt ; Natchaphon Burapanonte ; Wisarute Gunjarueg — People identification and counting system using raspberry Pi (AU-PiCC: Raspberry Pi customer counter)
DOI: 10.1109/ELINFOCOM.2016.7563020
Available: https://ieeexplore.ieee.org/document/7563020

[7] Yi-Chen Lee; Ching-Min Lee — Real-Time Smart Home Surveillance System of Based on Raspberry Pi
DOI: 10.1109/ECICE50847.2020.9301929
Available: https://ieeexplore.ieee.org/document/9301929

[8] Md. Tobibul Islam; Mohiuddin Ahmad; Akash Shingha Bappy — Real-Time Family Member Recognition Using Raspberry Pi for Visually Impaired People
DOI: 10.1109/TENSYMP50017.2020.9230937
Available: https://ieeexplore.ieee.org/document/9230937

[9] V. Sandeep; K. Lalith Gopal; S. Naveen; A. Amudhan; L. S. Kumar — Globally accessible machine automation using Raspberry pi based on Internet of Things
DOI: 10.1109/ICACCI.2015.7275764
Available: https://ieeexplore.ieee.org/document/7275764

[10] Mochammad Ariyanto; Ismoyo Haryanto; Joga Dharma Setiawan; M. Munadi; M. Sri Radityo — Real-Time Image Processing Method Using Raspberry Pi for a Car Model
DOI: 10.1109/ICEVT48285.2019.8993866
Available: https://ieeexplore.ieee.org/document/8993866

[11] D. Si; R. M. Edwards; Y. Geng — XBee Latency Analysis for Drone Mounted Machine Control over Wireless Communication Channels
DOI: 10.23919/EuCAP51087.2021.9411478
Available: https://ieeexplore.ieee.org/document/9411478

[12] SV Sriraam; Suraj Sajeev; Rohit Joshi; Amogh Vithalkar; Manish Bansal; Harshan Jagadeesh — Implementation of 5G Authentication and Key Agreement Protocol on Xbee Networks
DOI: 10.1109/COMSNETS48256.2020.9027314
Available: https://ieeexplore.ieee.org/document/9027314

[13] Shakila Baskaran; Themrichon Tuithung — Remote Monitoring and Control of Smart Distribution Grid Using Xbee Communication
DOI: 10.1109/ICCTCT.2018.8550890
Available: https://ieeexplore.ieee.org/document/8550890