Turkish Online Journal of Qualitative Inquiry (TOJQI)

Volume 12, Issue 3, June 2021: 2267-2275

Research Article

REAL TIME HUMAN MOVEMENT DETECTION USING RASPBERRY-PI

S.Jagadeesh Babu[1] S A. Aadarsh Praveen[2] Bharath G[3] Bhuvanesh DL[4] J.Yogapriya[5]

Abstract

The main purpose of our project is to enhance some issues in security camera. As these days security among the society is going down. Normally our security cameras will record the videos of thefts happening at night and we will see the video only the next day after realising that theft had happened. By this time we cannot stop it. So our project overcomes this issue by sending an alarm to the owner of the shop and thus the incident can be stopped at the time of happening.

This can be done by using a raspberry pi and a pi camera. The source code is implemented in raspberry pi and a pi camera is attached to it. When a human face is detected the pi camera takes a screenshot and it gives an alarm to the owner of the shop. The number of persons and time is noted at the time of incident.

The source code language used is python and the algorithm followed is Open CV.

Keyword: Raspberry Pi, OpenCV

Introduction

Real-time crowd counting in videos is becoming increasingly important for public area monitoring for safety and security. Crowd counting attempts to estimate the number of people passing through a given line or area. It can be used to monitor the number of people entering and

[1]Assistant Professor., Department of Electronics and Communication Engineering R.M.K. Engineering College, Chennai, India.sjb.ece@rmkec.ac.in

[2]B.E. Student, Department of Electronics and Communication Engineering R.M.K. Engineering College, Chennai, India. aada17101.ec@rmkec.ac.in

[3]B.E. Student, Department of Electronics and Communication Engineering R.M.K. Engineering College, Chennai, India. bhar17117.ec@rmkec.ac.in

[4]B.E. Student, Department of Electronics and Communication Engineering R.M.K. Engineering College, Chennai, India. bhuv17123.ec@rmkec.ac.in

[5]Program Analyst, CTS, Chennai. yogapriya.j@cognizant.com

exiting in a venue, estimate the flow of people in a subway station, and many other useful real-world applications.

This job still has a lot of obstacles to overcome. First, occlusion between people is a serious issue in crowded situations. Second, since the video resolution of surveillance cameras is poor, accurate information is lost. The majority of monitors in real-world locations, such as subway stations and libraries, are above people's heads.This is because cameras can catch the faces, dresses, and other characteristics of people walking by from this vantage point.

So, in our case, we'll assume that the cameras are mounted high and facing the direction of the crowd movement. The open CV definition is used to identify a human face. If a human face is detected in a given period of time, the pi camera that we have installed detects the person (that it is a human) and sends it to the appropriate person's email.

Literature Survey

1. One of the Raspberry Pi applications, according to Fatma Salih of Mysoon S.A. Omer, is to use it as a video server. Monitoring employs real-time video servers. The Raspberry Pi programming and control capabilities are used to create an embedded LAN live real-time video/audio stream server in this article. The video is taken via the Raspberry Pi camera module port and compressed before being sent over the network using a special standard that uses HTTP.

2. According to BrankoBalon and Milenko Simic, the Raspberry Pi is a versatile device the size of a credit card. Students have learned how to instal and use the Raspbian operating system, network and link to the Internet, write Python programmes, and create hardware-based projects.

3. According to Ishita Gupta, Varsha Patil, Chaitali Kadam, and Shreya Dumbre, implementing a Raspberry Pi-based face recognition system that uses traditional face detection and recognition techniques like Haar detection and PCA aims to make the system cost efficient, easy to use, and high performing.

4. Tsong-Yi Chen, Chao-Ho Chen, Da-Jinn Wang, and Yi-Li Kuo reported that after face detection, an individual is monitored by following the detected face, and then counted if their face touches the counting line.

5. Hanna Mohsin Ahmed and Rana Talib Rasheed claimed that they used OpenCV and Raspberry Pi to recognise faces.

Raspberry Pi

The Raspberry Pi Foundation produces a line of single-board computers known as Raspberry Pi. The Raspberry Pi is a low-cost computer that includes a series of GPIO pins for controlling electronic components and experimenting with the Internet of Things (IoT).

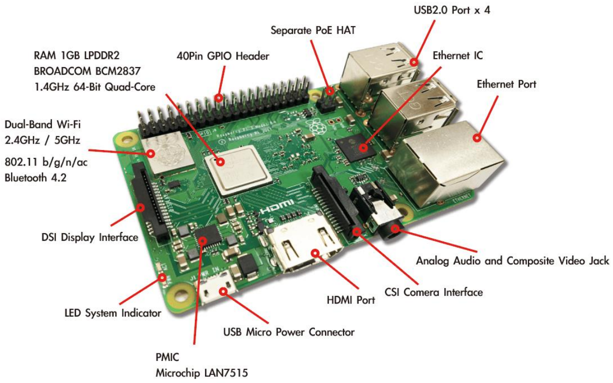
The Raspberry Pi runs on Linux, and its main supported operating system, Pi OS, runs a suite of open source applications. The Raspberry Pi Foundation contributes to the Linux kernel and other open source projects, as well as releasing open source versions of most of its own applications.

Raspberry pi 3b+

Specification:
The Raspberry Pi 3 Model B+ has the following specifications.

- Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- 1GB LPDDR2 SDRAM
- 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
- Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)
- Extended 40-pin GPIO header
- Full-size HDMI
- 4 USB 2.0 ports
- CSI camera port for connecting a Raspberry Pi camera
- DSI display port for connecting a Raspberry Pi touchscreen display
- 4-pole stereo output and composite video port
- Micro SD port for loading your operating system and storing data
- 5V/2.5A DC power input
- Power-over-Ethernet (PoE) support (requires separate PoE HAT)

Raspberry pi 3b+ board



Pi Camera

The Pi camera module is a small, light camera that works with the Raspberry Pi. The MIPI camera serial interface protocol is used to communicate with the Pi. It's commonly used in image recognition, machine learning, and surveillance applications.

The Raspberry Pi Camera Features

- Fully Compatible with Both the Model A and Model B Raspberry Pi
- 5MP Omnivision 5647 Camera Module
- Still Picture Resolution: 2592 x 1944
- Video: Supports 1080p @ 30fps, 720p @ 60fps and 640x480p 60/90 Recording
- 15-pin MIPI Camera Serial Interface - Plugs Directly into the Raspberry Pi Board
- Size: 20 x 25 x 9mm
- Weight 3g
- Fully Compatible with many Raspberry Pi cases

Raspberry pi camera board module

Open CV

OpenCV is a cross-platform library that can be used to build real-time computer vision apps. It focuses primarily on image processing, video recording, and analysis, with features such as face detection and object detection. The creation of explicit, meaningful representations of physical objects from their images is known as computer vision.A depiction or representation of structures in a 3D scene is the performance of computer vision.

Security Application

Surveillance using biometrics (iris, finger print, facial recognition) to detect suspicious activities or behaviours

Features of OpenCV Library

- You can read and write images using the OpenCV library.

- Video capture and storage
- Image processing (filter, transform)
- Make a function detection.
- Identify individual objects in videos or photographs, such as faces, eyes, and vehicles.
- Analyze the video by estimating its motion, removing the backdrop, and tracking the objects in it.

Open CV flowchart

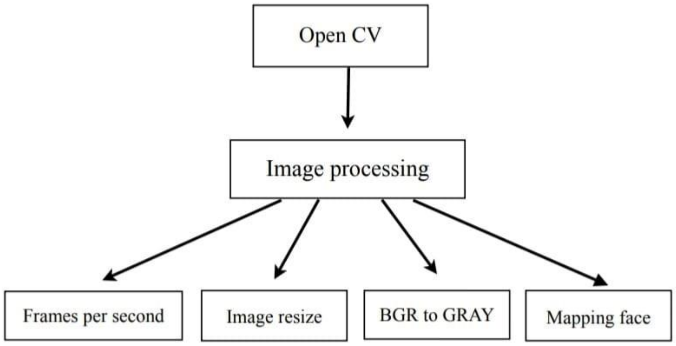


Image processing

Image processing is a technique for applying operations to an image in order to improve it or extract useful information from it. It's a form of signal processing in which the input is an image and the output is either that image or its characteristics/features. This method accepts a video segment or an image, such as a picture, as input. The performance matches the desired or attention-getting part of the picture. In this project we are using image processing to resize the image and convert image from BGR to Grey and to map the co-ordinates for the face and to convert videos to frames per second.

1. Frames per second

The input videos are converted to images using frames per second method as a result the images are obtained to detect faces.

fps = FPS().start()

2. **BGR to GRAY:**

Convert the input frame from BGR to grayscale for face detection.

```
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

3. **Resize the image:**

The captured image after fps is being resized for the required amount for face detection.

```
frame = imutils.resize(frame, width=500)
```

4. **Mapping of face:**

After converting from BGR to grey image, the image is sent for face detection the face detection is done by mapping the boundaries (top, left, right, bottom) the co-ordinate axis create a box around the face.

```
boxes = [(y, x + w, y + h, x) for (x, y, w, h) in rects]
```

Face detection

There is a face detection package in python that helps to locate the face in an image. The image is first imported by specifying the image's location. The image is then converted from RGB to Grayscale since faces are easier to identify in grayscale. After that, image editing is used, which includes resizing, cropping, blurring, and sharpening if necessary. Image segmentation is the next step, which is used for contour detection or to segment several objects in a single image so that the classifier can easily detect the objects and faces in the image. The next move is to use the Python face detection package. The position of human faces in a picture or image is determined using this algorithm. The face detection package in Python is used to select or extract features for an object in an image using edge detection, line detection, and centre detection for detecting eyes, noses, mouths, and other features in the image. It's used to pick out the most important features in an image and remove them for face detection. The next step is to enter the x, y, w, and h coordinates, which will create a rectangle box in the image to indicate the position of the face. It can then draw a rectangular box in the field of interest where the face is detected. When the face is detected the image is sent to IoT cloud storage.

The face detection package used is **importface_recognition**

IoT cloud storage

When the pi camera captures images continuously when a face is detected the raspberry pi needs to store the images and send it to the respective person’s mail. In order the store the images IoT cloud storage is used, the capture image is sent to the person’s mail. The time, date and count of person’s are stored in the cloud we can access the cloud in order to see the time, date and count. A cloud has a larger memory and the details are noted in the cloud accurately and it can store as many images it wants. Since the raspberry pi ram is small we are using cloud storage.

Mail

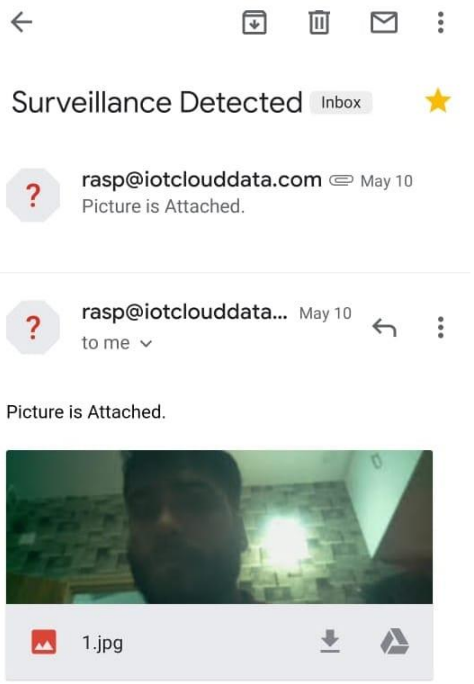
The screenshot of the image captured send to mail.

```
import email
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
from email.mime.base import MIMEBase
from email import encoders
```

These are the packages used for send the image to mail

```
sender = 'rasp@iotclouddata.com'
password = 'xyz'
receiver1 = 'xyz@gmail.com'
```

The iot cloud storage is the sender’s address from there the image is being sent to the owner’s mail id.



Surveillance Detected Inbox

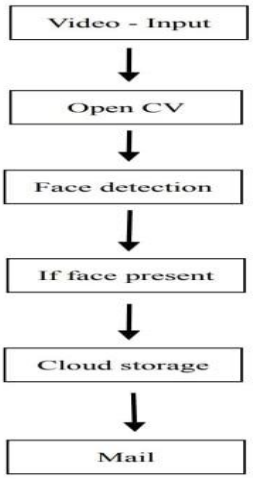
rasp@iotclouddata.com May 10
Picture is Attached.

rasp@iotclouddata... May 10
to me

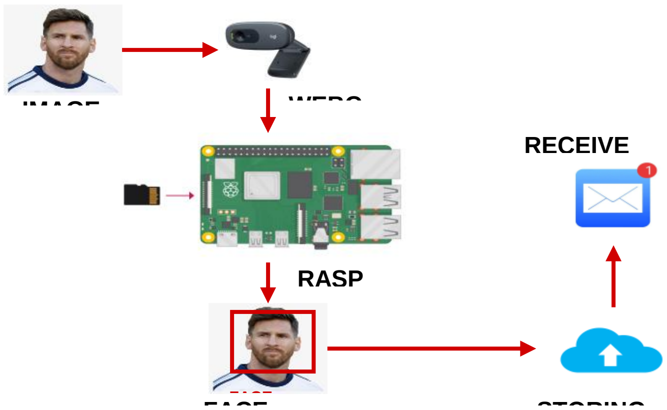
Picture is Attached.

1.jpg

Flowchart of model



Methodology



- Connect SD card with OS to Raspberry Pi.
- Connect the webcam to Raspberry Pi and download and import necessary packages.
- Setup OpenCV on Raspberry Pi.
- Connect the webcam to Raspberry Pi and download and import necessary packages.
- Import face detection package for frontal face detection.
- Take image input through the webcam.
- Detect the four corners of the face from the image.
- Draw a square over the image with the received coordinates.
- Store image , number of faces detected and time at which image was taken on to the cloud.
- Send email notification with face detected image.

Conclusion

The project performs the given condition the face of a person is recognised and the number of persons in particular frame are counted. The output is sent to the person's mail and the person can find out the time of theft by logging to the server. The server notes the time and the count of person's also. The accuracy of the project is 95%.

Reference

1. Fatma Salih, Mysoon S.A. Omer, "Raspberry pi as a Video Server", International Conference on Computer, Control, Electrical and Electronics Engineering(ICCCEEE),2018.

2. BrankoBalon, Milenko Simic, “Using Raspberry pi Computers in Education”, 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO),2019.

3. Ishita Gupta, Varsha Patil, Chaitali Kadam, Shreya Dumbre, “Face Detection and Recognition Using Raspberry Pi”, IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE),2016.

4. Tsong-Yi Chen, Chao-Ho Chen, Da-Jinn Wang, Yi-Li Kuo, “People Counting System based on Face Detection”, Fourth International Conference on Genetic and Evolutionary Computing,2010.

5. Hanaa Mohsin Ahmed, Rana Talib Rasheed, “A Raspberry pi Real-Time Identification System on Face Recognition”, 2020 1st. Information Technology To Enhance e-learning and Other Application (IT-ELA),2020.

6. M. Abdulhamid, O. Odondi and M. AL-Rawi, "Computer Vision Based on Raspberry Pi System”,2020.

7. Pritish Sachdeva and Shrutik Katchii, "A Review paper on Raspberry Pi", Proceedings International Journal of Current Engineering and Technology IJCET,2014.

8. W. Zhao, R. Chellappa, P. J. Phillips and A. Rosenfeld, "Face Recognition: A Literature Survey", ACM ComputingSurveys, 2003.

9. Jafri R. and Arabnia H. R., "A survey of face recognition techniques, " Journal of Information Processing Systems,2009.

10. Bharadwaj Vishanth Thiyagarajan, "Face Detection and Facial Feature Points' Detection with the help of KLT Algorithm", IJARCSMS, , 2014.