Implementation of Video Surveillance Using Raspberry PI 2

SK.Nabeena,
PG student,
Department of ECE,
Sri Vasavi Engineering College,
Tadepalligudem, India.

N.V.Narayana Rao,
Sr. Assistant Professor,
Department of ECE,
Sri Vasavi Engineering College,
Tadepalligudem, India.

ABSTRACT

In this paper, a PIR based advanced video surveillance capable of face detection and at the same time recognizing the face detected is implemented. Here a system is developed using a PIR sensor and webcam interfaced to raspberry pi 2 to perform video surveillance. Face detection and Face recognition is coded using Python scripting language. PIR (Passive Infra Red) sensor is a pyroelectric device that detects motion by measuring changes in the infrared levels emitted by surrounding objects in a specific area. According to the change in infrared radiation, there will be change in voltages generated which are amplified and used to turn ON the webcam for video monitoring at the same time face detection and face recognition is also done in surveillance area. Thus it can save power consumption and the memory space of the recording system as webcam will only get turned ON when PIR sensor detects a human.

Keywords: Raspberry Pi 2, PIR Sensor, Python, OpenCV, WebCam, Haar-Training, XML file, Face Recognition.

1. INTRODUCTION

Security systems have been widely implemented in the world. There are lots of security systems in the market for both indoor and outdoor applications such as ultrasonic detectors, CCTV, microwave detectors, photoelectric detectors, infrared detectors etc. However those are expensive, have more electrical power consumption, more memory space utilization of the recording system.

A solution to overcome these problems could be by using a sensor of low cost (Passive Infra Red sensor) which has the ability to detect the humans as they come within the sensor's detection range and generates an output. This output can be used to make an surveillance camera with face recognition. For achieving this task Raspberry Pi2 hardware with Raspbian OS was chosen. The visual input is taken by USB camera connected to raspberry pi2 via USB port of pi.

The whole code for PIR based advanced video surveillance is written in python using OpenCV libraries. When an human comes in the detection range of PIR sensor, the system is programmed to function for face recognition. For this bootable scripts are used to run the program for face recognition.

Once a face is captured in a frame, then it is matched against the trained database on pi. If the face is recognized as a trained one then name of the person with hit rate is shown.

2. IMPLEMENTATION DETAILS

The implementation details on hardware shown in Figure 1 and software used in the project:

Figure1. Hardware implementation

2.1 Hardware: Raspberry Pi 2

Raspberry pi 2 [2]is a single board computer whose size is much smaller than PC. The Raspberry Pi 2 delivers 6 times the processing capacity of previous models. This second generation Raspberry Pi has an upgraded Broadcom BCM2836 processor, which is a powerful ARM Cortex-A7 based quad-core processor that runs at 900MHz. The board has RAM memory capacity of 1Gbyte.It has four USB ports, HDMI port and 10/100 BaseT Ethernet socket. A micro-USB connector as power supply, which produce at least 2A at 5 volts. The standard USB keyboards and mouse need to connect with the Raspberry Pi2. It works on Raspbian OS[3].

2.2 Passive Infrared Sensor (PIR)

The PIR sensor [6] is the core part of the system. Everything on earth has infrared (IR) energy. The IR energy of an object depends on different parameters such as its temperature, color, and texture. This energy is not visible with the human eye, but PIR sensors can detect it. The system basically functions on infrared radiation, which is emitted from human body. The PIR (Passive Infra-Red) Sensor [7] is a pyroelectric device that detects motion by measuring changes in the infrared levels emitted by surrounding objects. This motion can be detected by checking for a high signal on a single I/O pin.

PIR sensor is widely used in security system to detect the motion of human. Pyroelectric devices, such as the PIR sensor, have elements made of a crystalline material that generates an electric charge when exposed to infrared radiation. The changes in the amount of infrared striking the element change the voltages generated, which are measured by an on-board amplifier. The device contains a special filter called a Fresnel lens, which focuses the infrared signals onto the element. As the ambient infrared signals change rapidly, the on-board amplifier trips the output to indicate motion. The PIR Sensor [8] has a range of approximately 20 feet.

2.3 WEB CAMERA

QHMPL camera with Image Resolution of 25 Mega Pixels is used. It is interpolated with 6 white lights used to ON when in dark. It has high quality CMOS sensor image control color saturation, brightness, sharpness and brightness is adjustable. These web cameras continuously monitor the room and send the video. The webcam comes equipped with USB 2.0 interface and has a maximum frame rate of 30fps. It sports a focus range of 4-cm to infinity and has an integrated microphone.

SOFTWARE
2.4 Open CV:

OpenCV i.e. Open Source Computer Vision[4] is a library of programming functions that helps in real-time image processing. OpenCV was designed for computational efficiency and with a strong focus on real time applications. OpenCV is written in optimized C and can take advantage of multicore processors. OpenCV's goals are to provide a simple-to-use computer vision infrastructure that helps people build fairly sophisticated vision applications quickly.

For image capturing and image processing OpenCV was used on Raspberry Pi. And also Haar feature-based Cascade Classifiers for face detection.

2.5 Python:

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python was designed to be highly readable which uses English keywords frequently where as other languages use punctuation and it has fewer syntactical constructions than other languages. Python [5] is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. The Python interpreter is easily extended with new functions and data types implemented in C or C++ (or other languages callable

from C). Python is also suitable as an extension language for customizable applications.

3. TRAINING METHOD

For training face need to create a cascade of classifiers based on Haar-like features,[9] which is the most common technique in computer-vision for face detection [10].

Training Steps to Create a Haar-like Classifier:
STEP 1: Collecting Image Database

Collect 50 positive sample images for training. The positive images are those images that contain the object (e.g. human).

STEP 2: Capture & Mark Positive Images

In this step need to create a data file (vector file) that contains the names of positive images as well as the location of the objects in each image.

STEP 3: Creating a vector of positive images

In folder..\training\ there is a batch file named people.txt The content of the batch file is:People.txt - info positive/info.txt -vec vector/facevector.vec -num 50 -w 100 -h 100The batch file loads info.txt and packs the object images into a vector file with the name of e.g. facevector.vec After running the batch file, will have the file facevector.vec in the folder..\training\vector

STEP 4: Haar-Training

The OpenCV create samples-utility outputs a vector with samples, this vector is used as input to the OpenCV haartraining utility.

Main Parameters:-info positive/info.txt Path for positive info file-vec vector/facevector.vec Path for the output vector file-num 50 Number of positive files to be packed in a vector file-w 100 Width of objects -h 100 Height of objects

STEP 5: Creating the XML File

After finishing Haar-training step, in folder../training/cascades/ have catalogues named from "0" upto "N-1" in which N is the number of stages already defined in haartraining.bat. In each of those catalogues there should be HaarClassifier.txt file.

Copy all the folders 0..N-1 into the folder ../cascade2xml/data/ Now combine all created stages (classifiers) into a single XML file which will be final file a "cascade of Haar-like classifiers". Run the batch file people.txt at ../cascade2xml/ which is: haarconv.exe data myfacedetector.xml 100 100 myfacedetecor.xml is the output file name and 100 100 are W and H respectively. Now have your own XML file. Copy it in MyCascade folder, point to this classifier from project source code, and run face recognition program.

4. FACE RECOGNITION:

Face recognition is an essential task in today`s world with various types of applications such as surveillance purposes, monitoring applications, security application. Fisher faces method is one of the important method in face recognition.

4.1. FISHERFACES METHOD:

Fisher faces is used to identify faces, by training and testing with several faces under different lighting. Fisher Linear Discriminant (FLD) analysis, also called Linear Discriminant Analysis (LDA) [11] finds the line that best separates the points. For example, consider two sets of points, colored green and blue, in two-dimensional space being projected onto a single line. Depending on the direction of the line, the points can either be mixed together or be separated. In terms of face recognition this means grouping images of the same class and separate images of different classes. Images are projected from a N-dimensional space, where N is the number of pixels in the image, to a M-1 dimensional space, where M is the number of classes of images.
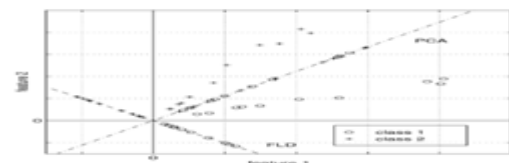

Fig.a. linear discriminant analysis

The LDA method, which creates an optimal projection of the dataset, maximizes the ratio of the determinant

of the between-class scatter matrix of the projected samples to the determinant of the within-class scatter matrix of the projected samples. The within-class scatter matrix, also called intra-personal, represents variations in appearance of the same individual due to different lighting and face expression, while the between-class scatter matrix, also called the extra-personal, represents variations in appearance due to a difference in identity. In this way, fisher faces can project away some variation in lighting and facial expression while maintaining discrimination ability. With LDA, it is possible to classify the training set to deal with different people and different facial expressions [12].

4.2. FACE RECOGNITION PROCESSING

Feature extraction is performed to provide effective information that is useful for distinguishing between faces of different persons. For face matching, the extracted feature vector of the input face is matched against those of trained faces in the database; output identity the face with name of the trained person with hit rate when a match is found. The number of training stages with minimal desired hit rate will determine the total hit as hit rate of 99:9% would yields as hit rate of 0:999 = 0:98 though the cascade.

5. DESIGNING THE SURVEILLANCE SYSTEM

In this paper, an advanced video surveillance capable of face detection and at the same time recognizing the face detected was developed. The main devices used in this system are: PIR for sensing the object, web camera for capturing the data, and raspberry pi 2 for processing the data.

5.1. System Architecture

In more details, the system can be described in Figs. 2 and 3 (in the form of flowchart, algorithm). The system works as follows.
a. The PIR sensor which has been integrated with raspberry pi 2 through socket GPIO, detects an object

(human) according to the temperature of the object and is captured by webcam using the OpenCV library.
b. When PIR sensor sense an object the system is programmed to function for face recognition. For this bootable scripts are used to run the program for face recognition.
c. Once a face is captured in a frame, then it is matched against the trained database on pi.
d. If the face is recognized as a trained one then name of the person with hit rate is shown.
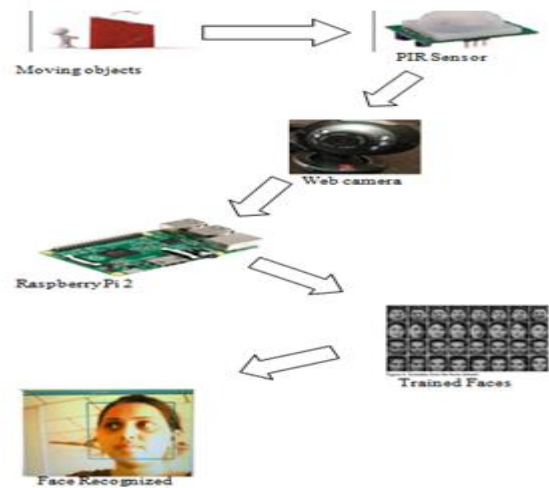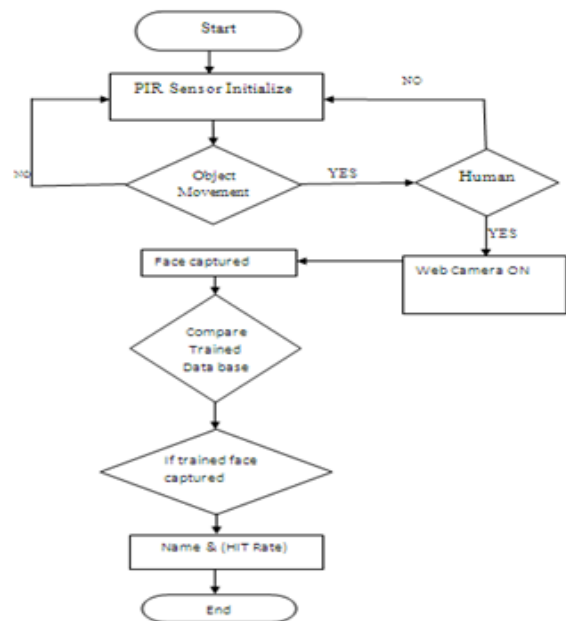

Fig3: Data flow

Fig4: Algorithm

6. RESULT

The experimental result shows that the system works satisfactory if the distance of the object (human) is less than 5m. The PIR sensor is able to detect an object up to 7m successfully. When the person comes within the sensor's detection range the face recognition process executes and the result is displayed i.e. name of the trained face with hit rate. Examples of the trained and recognized faces on web camera are shown in Figs. 4 and 5, respectively

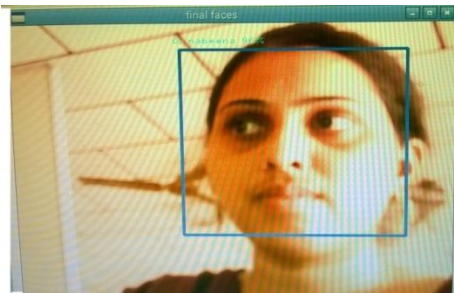
Fig 5: Trained Faces on database


Fig6: Face Recognized based on trained database.

7. CONCLUSION

In this paper, a PIR based advanced video surveillance capable of face detection and at the same time recognizing the face detected is proposed. The proposed system can save the memory space of the recording system as it starts recording when the webcam is turned ON, only when PIR sensor gets activated. Both webcam and PIR sensor gets activated only when there is an intruder in the detection range of the PIR sensor. The low cost PIR sensor is easy to interface with other components. By using this system the power consumed and memory space of the system are reduced. Based on the experimental result, the system has been able to capture an object (human), and recognized at PIR detection range. In future, this surveillance system can be enhanced by replacing the sensors to the more advanced ones, such that it is able to detect an object whose distance is more than 7

meters. Furthermore, the resolution of the captured image should also be increased to make it easier to manually recognize the object in the image.

REFERENCES

[1]. "FLIR Portable Thermal Handheld Surveillance Camera," Costco Wholesale, [Online]. Available: http://www.costco.com/.product.100095497.html? cm_sp=RichRelevance-_-categorypage Horizontal Top-_CategoryTopProducts&cm_vc=category pageHorizontalTop|CategoryTopProducts.

[2]. "What is a Raspberry Pi?" Raspberry Pi Foundation, [Online]. Available: http://www.raspberrypi.org/.

[3].Installing Raspbian - http://www.raspberrypi.org/help/noobs-setup/

[4].OpenCV, "Open Source Computer Vision Library," opencv dev team, 21 April2014.[Online]. Available:http://docs.opencv.org/ modules/core/doc/ intro.html.

[5] python documentation http://www.python.org/doc Language Reference http://www.python.org/doc/current/ref/ref.html

[6]. Adafruit, "PIR Motion sensor," Adafruit, 30 August 2014. [Online]. Available: https://learn.adafruit.com/pir-passive- infrared proximity-motion sensor/overview.

[7] Herbert L. Berman and Los Altos Hills, "Infrared Intrusion Detector System ," 3,73,718, Nov 21, 1972.

[8] L. Herbert, Berman, Altos Hills Los , and Calif, "Infrared Intrusion Detector System," 3,703,718, Nov. 21, 1972.

[9] Mahdi Rezaei, Reinhard Klette, "3D cascade of classifiers for open and closed eye detection in
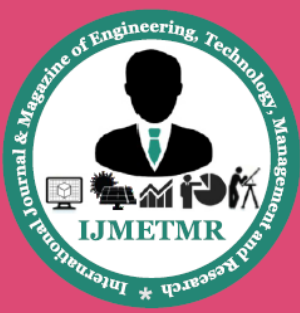
driver distraction monitoring", In Proc. Computer Analysis of Images and Patterns, pp. 171-179, 2011.

[10] M. Guillaumin, T. Mensink, J. J. Verbeek, and C. Schmid, "Automatic face naming with caption-based supervision," in Proc. CVPR, 2008, pp. 1–8.

[11] R. Chellappa, C. Wilson, and S. Sirohey, "Human and Machine Recognition of Faces: A Survey," Proc. IEEE, vol. 83, no. 5, pp. 705-740, 1995.

[12] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," European Conf. Computer Vision, 1996, pp. 45-58.