



**THE DETECTION AND RECOGNITION OF  
FACES IN THE INTERNET OF THINGS  
FOR SECURITY APPLICATIONS**

**NASHWAN ADNAN OTHMAN  
(152129109)**

**MASTER THESIS**

**Department: Computer Engineering  
Supervisor: Assoc. Prof. Dr. İlhan AYDIN**

**JANUARY – 2018**

**REPUBLIC OF TURKEY**  
**FIRAT UNIVERSITY**  
**THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE**

**THE DETECTION AND RECOGNITION OF FACES IN THE INTERNET OF  
THINGS FOR SECURITY APPLICATIONS**

**MASTER THESIS**  
**NASHWAN ADNAN OTHMAN**  
**(152129109)**

**Department of Computer Engineering**  
**Submission Date to the Institute: 22 December 2017**  
**Thesis Presentation Date: 15 January 2018**

**Thesis Supervisor: Assoc. Prof. Dr. İlhan AYDIN (F. U.)**

**Other Juries: Assoc. Prof. Dr. Mehmet KARAKÖSE (F. U.)**

**: Assoc. Prof. Dr. Muhammed Fatih TALU (I. U.)**

**JANUARY – 2018**

## **ACKNOWLEDGMENT**

First, I give thanks to God for protection and ability to do my work. I would like to express my sincere gratitude to my supervisor Assoc. Prof. Dr. İlhan AYDIN for his patience, kind support, immense knowledge, motivation, directions and thorough guidance during my research work. His guidance helped me during the time of research. At many stages of this project, I benefit from his advice, particularly when exploring new ideas. His positive evaluating and confidence in my research inspired me and gave me confidence. His careful editing contributed enormously to the production of this thesis.

I would like to thank all my friends, who have supported me throughout the entire process, both by keeping me harmonious and helping me putting pieces together. Your friendship makes my life a wonderful experience. I cannot list all of their names here. I will be grateful forever for their kindness.

Last but not the least, I have to thank my parents for their love, encouragement, prayer, and support throughout my life. Thank you both for giving me strength to reach the stars and chase my dreams.

**Nashwan Adnan OTHMAN**

**ELAZIG - 2018**

## TABLE OF CONTENTS

|                                     | Page No |
|-------------------------------------|---------|
| ACKNOWLEDGMENT.....                 | II      |
| TABLE OF CONTENTS.....              | III     |
| ABSTRACT.....                       | VI      |
| ÖZET .....                          | VII     |
| LIST OF FIGURES .....               | VIII    |
| LIST OF TABLES .....                | X       |
| ABBREVIATIONS .....                 | XI      |
| 1. INTRODUCTION .....               | 1       |
| 1.1 A Literature Review .....       | 4       |
| 1.2 The Aim of Thesis.....          | 5       |
| 1.3 Organisation of the Thesis..... | 6       |
| 2. STATE OF ART .....               | 7       |
| 2.1 Internet of Things .....        | 7       |
| 2.1.1 Top IoT Applications .....    | 8       |
| 2.1.2 IoT Platform .....            | 10      |
| 2.1.3 IoT Architecture .....        | 10      |
| 2.2 Computer Vision .....           | 11      |
| 2.3 Hardware Implementation.....    | 12      |
| 2.3.1 Raspberry Pi 3 .....          | 12      |
| 2.3.2 Raspberry Pi Camera .....     | 14      |
| 2.3.3 PIR Sensor .....              | 15      |
| 2.3.4 Smartphone .....              | 18      |
| 2.3.5 SD Card Memory .....          | 19      |
| 2.3.6 Power Supply.....             | 19      |

|           |                                                                                        |           |
|-----------|----------------------------------------------------------------------------------------|-----------|
| 2.4       | Software Implementation.....                                                           | 19        |
| 2.4.1     | Raspbian OS .....                                                                      | 19        |
| 2.4.2     | Putty Configuration .....                                                              | 20        |
| 2.4.3     | VNC Server .....                                                                       | 21        |
| 2.4.4     | Python Languages.....                                                                  | 22        |
| 2.4.5     | OpenCV Library .....                                                                   | 23        |
| 2.4.6     | Telegram Bot API.....                                                                  | 23        |
| 2.5       | System Design .....                                                                    | 25        |
| <b>3.</b> | <b>A NEW IOT BASED FACE DETECTION BY USING HAAR-LIKE FEATURES</b>                      | <b>26</b> |
| 3.1       | The Proposed Method for Face Detection.....                                            | 27        |
| 3.1.1     | Integral Images .....                                                                  | 28        |
| 3.1.2     | Haar-Like Features .....                                                               | 29        |
| 3.1.3     | Adaboost Training.....                                                                 | 30        |
| 3.1.4     | Cascade Classifier .....                                                               | 30        |
| 3.2       | Experimental Results.....                                                              | 31        |
| <b>4.</b> | <b>AN IOT BASED FACE RECOGNITION FOR FOREIGN PERSON<br/>DETECTOR .....</b>             | <b>37</b> |
| 4.1       | The Proposed Face Recognition Method .....                                             | 38        |
| 4.1.1     | Face Representation .....                                                              | 39        |
| 4.1.2     | Feature Extraction .....                                                               | 39        |
| 4.1.3     | Classification.....                                                                    | 43        |
| 4.2       | Experimental Results.....                                                              | 44        |
| <b>5.</b> | <b>HOG BASED BODY DETECTION AND ITS IOT IMPLEMENTATION<br/>ON RASPBERRY PI 3 .....</b> | <b>48</b> |
| 5.1       | The Proposed Body Detection Method .....                                               | 49        |
| 5.2       | Experimental Results.....                                                              | 53        |
| <b>6.</b> | <b>CONCLUSIONS .....</b>                                                               | <b>60</b> |

|                             |           |
|-----------------------------|-----------|
| <b>REFERENCES.....</b>      | <b>61</b> |
| <b>CURRICULUM VITA.....</b> | <b>68</b> |



## ABSTRACT

### THE DETECTION AND RECOGNITION OF FACES IN THE INTERNET OF THINGS FOR SECURITY APPLICATIONS

In recent years, the security constitutes are the most important section of human life. Security of the house and the family is important for everybody. Automation of a home is an exciting field for security applications. This area has developed with new technologies such as the Internet of things (IoT). In IoT, each device behaves as a small part of an internet node and each node communicates and interacts. Currently, security cameras are used in order to construct safety in areas, cities and homes. The camera records the events, and when a problem occurs, it will detect by monitoring the old recording. At this time, the cost is the greatest factor. This system is very helpful to reduce the cost of monitoring the movement from outside. The purpose of this thesis is to describe a security alarm application by utilizing low preparing power chips and Internet. Also, a new online method is proposed to detect and recognize faces on Raspberry Pi in the IoT. Raspberry Pi operates and controls movement detectors. It will monitor and record the motions for future playback. This thesis proposes an analysis of images via computer vision to detect and recognize faces in the analyzed images. If these frames contain a face, the system will detect and recognize the face. This system is appropriate for small personal range surveillance, as, in personal office security, home, parking entrance and bank locker room. The face detection and recognition in the IoT is a very important problem for a security and surveillance system. Also, face detection and recognition is presently a very active research area. The proposed system is very helpful to reduce the cost of monitoring the movement from outside.

On the other hand, in this research, an IoT-based system is combined with computer vision in order to detect human's body. A Raspberry Pi 3 cards with the size of a credit card is used for this purpose. A motion is detected by the PIR sensor mounted on the Raspberry Pi. PIR sensor helps to monitor and get alerts when a movement is detected. Afterward, the human's body detects in the captured image and is sent to a smartphone by using telegram application.

**Keywords:** Internet of things, Security, Computer Vision, Raspberry Pi 3, PIR sensor, Smartphone, Face detection, Face recognition, Human detection, Telegram.

## ÖZET

### GÜVENLİK UYGULAMALARI İÇİN NESNELERİN İNTERNETİNDE YÜZLERİN TESPİTİ VE TANINMASI

Son yıllarda, güvenlik konusu insan yaşamının en önemli parçası olmuştur. Ev ve aile güvenliği herkes için önemlidir. Akıllı ev otomasyonu güvenlik uygulamaları için heyecan verici bir alandır. Bu alan nesnelerin interneti gibi yeni teknolojiler ile gelişmiştir. Nesnelerin internetinde her aygıt bir internet düğümünün küçük bir parçası olarak davranır ve her düğüm birbiri ile iletişim kurar ve etkileşir. Mevcut olarak, güvenlik kameraları evler, şehirler ve alanlardaki güvenliği oluşturmak için kullanılır. Kameralar olayları kaydeder ve bir problem olduğunda eski kayıtlar izlenerek tespit edilir. Bu yöntem kullanıldığında maliyet en önemli faktördür. Tezin amacı düşük güçlü bir çip ve internetten faydalanarak bir güvenlik alarm sistemi oluşturmaktır. Aynı zamanda Rasbery PI ve nesnelerin interneti üzerinde yüz tespiti ve tanıma için yeni bir çevrimiçi sistem önerilmektedir. Rasbery PI hareket olup olmadığını PIR sensör ile kontrol eder. İleride oynatılmak üzere hareketler izlenmekte ve kaydedilmektedir. Bu tez, analiz edilen görüntülerdeki yüzleri tespit etmek ve tanımak için bilgisayar görme yoluyla görüntü analizini önermektedir. Kameradan alınan çerçeveler bir yüz içeriyorsa, sistem yüzü algılar ve tanır. Bu sistem kişisel ofis güvenliği, ev, otopark girişinde ve banka kasa odasında küçük kişisel gözetim için uygundur. Nesnelerin internetinde yüz tespiti ve tanıma, bir güvenlik ve gözetim sistemi için çok önemli bir sorundur. Ayrıca, yüz tanıma ve tanıma, günümüzde çok aktif bir araştırma alanıdır. Önerilen sistem dışarıdaki hareketi izlem maliyetini azaltmak için oldukça kullanışlıdır.

Öte yandan, bu çalışmada, insan vücudunu algılamak için IoT tabanlı bir sistem bilgisayar görmesiyle birleştirildi. Bu amaçla, kredi kartı boyutunda bir Rasbery Pi 3 kartı kullanılmıştır. Rasbery Pi üzerine monte edilen PIR sensörü tarafından hareket olup olmadığı tespit edilmektedir. PIR sensörü, bir hareket tespit ettiğinde uyarıları izlemeye ve almaya yardımcı olur. Daha sonra insan vücudu çekilen görüntüde tespit edilir ve telegram uygulamasıyla bir akıllı telefona gönderilir.

**Anahtar Kelimeler:** Nesnelerin interneti, Güvenlik, Bilgisayar görmesi, Rasbery Pi 3, PIR sensörü, Akıllı telefon, Yüz algılama, Yüz tanıma, İnsan tespiti, Telegram.



## LIST OF FIGURES

|                                                                                             |    |
|---------------------------------------------------------------------------------------------|----|
| <b>Figure 2.1.</b> The application areas of IoT.....                                        | 9  |
| <b>Figure 2.2.</b> The IoTs architecture .....                                              | 11 |
| <b>Figure 2.3.</b> Raspberry pi 3 block diagram .....                                       | 13 |
| <b>Figure 2.4.</b> GPIO pins of Raspberry Pi 3.....                                         | 14 |
| <b>Figure 2.5.</b> Raspberry Pi cameras .....                                               | 15 |
| <b>Figure 2.6.</b> Connecting PIR sensor to Raspberry Pi .....                              | 16 |
| <b>Figure 2.7.</b> Sense motion by PIR sensor .....                                         | 17 |
| <b>Figure 2.8.</b> Possible detection range .....                                           | 18 |
| <b>Figure 2.9.</b> Installed Raspbian OS .....                                              | 20 |
| <b>Figure 2.10.</b> Putty configurations .....                                              | 21 |
| <b>Figure 2.11.</b> Installed VNC servers .....                                             | 22 |
| <b>Figure 2.12.</b> Created boot and display tokens.....                                    | 24 |
| <b>Figure 2.13.</b> System design.....                                                      | 25 |
| <b>Figure 3.1.</b> Flowchart for proposed system .....                                      | 26 |
| <b>Figure 3.2.</b> The pseudo code of integral image calculation and a numeric example..... | 28 |
| <b>Figure 3.3.</b> Different kinds of features .....                                        | 29 |
| <b>Figure 3.4.</b> Haar-like feature example .....                                          | 30 |
| <b>Figure 3.5.</b> Example for cascaded of phases.....                                      | 31 |
| <b>Figure 3.6.</b> Experimental setup for face detection application .....                  | 32 |
| <b>Figure 3.7.</b> Show face detection result on the computer screen .....                  | 33 |
| <b>Figure 3.8.</b> Screenshot of received notification on the smartphone .....              | 33 |
| <b>Figure 3.9.</b> Result when face is discovered .....                                     | 34 |
| <b>Figure 3.10.</b> Result when face not discovered.....                                    | 34 |
| <b>Figure 3.11.</b> Result when multiple face discovered.....                               | 35 |
| <b>Figure 4.1.</b> Flowchart for face recognition system .....                              | 37 |
| <b>Figure 4.2.</b> General steps for identification procedure with face recognition .....   | 39 |
| <b>Figure 4.3.</b> preprocessed image divided to 64 areas.....                              | 40 |
| <b>Figure 4.4.</b> Example of LBP calculation .....                                         | 41 |
| <b>Figure 4.5.</b> An example of extracting LBPH feature vector.....                        | 42 |
| <b>Figure 4.6.</b> Graphic diagram of KNN classification.....                               | 43 |
| <b>Figure 4.7.</b> Experimental setup for face recognition .....                            | 44 |

|                                                                                                    |    |
|----------------------------------------------------------------------------------------------------|----|
| <b>Figure 4.8.</b> Display output on the PC screen .....                                           | 45 |
| <b>Figure 4.9.</b> Screenshot of received notification on the smartphone for face recognition ...  | 46 |
| <b>Figure 4.10.</b> The output of the algorithm for different face detection and recognition ..... | 46 |
| <b>Figure 4.11.</b> The output of the algorithm for unknown face .....                             | 47 |
| <b>Figure 4.12.</b> Execution time comparison graph for face recognition algorithms .....          | 49 |
| <b>Figure 4.13.</b> Accuracy comparison graph for face recognition algorithms .....                | 50 |
| <b>Figure 5.1.</b> Flowchart for body detection system .....                                       | 51 |
| <b>Figure 5.2.</b> Preprocessing image and gamma normalization .....                               | 53 |
| <b>Figure 5.3.</b> Computing the gradients .....                                                   | 54 |
| <b>Figure 5.4.</b> Compute histogram of gradient orientation .....                                 | 55 |
| <b>Figure 5.5.</b> Experimental setup for body detection .....                                     | 56 |
| <b>Figure 5.6.</b> Display body detection output on the PC screen .....                            | 57 |
| <b>Figure 5.7.</b> Screenshot of received notification on the smartphone for body detection .....  | 58 |
| <b>Figure 5.8.</b> The output of the algorithm for one and multiple human detection .....          | 58 |

## LIST OF TABLES

|                                                                                                |    |
|------------------------------------------------------------------------------------------------|----|
| <b>Table 3.1.</b> Execution time and Detection Rate for face detection .....                   | 35 |
| <b>Table 4.1.</b> Execution time comparison for face recognition .....                         | 48 |
| <b>Table 4.2.</b> Detection rate comparison for face recognition.....                          | 49 |
| <b>Table 5.1.</b> The detection rate and execution time for body detection on Raspberry Pi 3.. | 59 |



## ABBREVIATIONS

|                |                                                 |
|----------------|-------------------------------------------------|
| <b>IOT:</b>    | Internet of Things                              |
| <b>PIR:</b>    | Passive Infrared                                |
| <b>LBP:</b>    | Local Binary Patterns                           |
| <b>LBPH:</b>   | Local Binary Pattern Histogram                  |
| <b>LDA:</b>    | Linear Discriminant Analysis                    |
| <b>PCA:</b>    | Principal Component Analysis                    |
| <b>HOG:</b>    | Histogram of Oriented Gradient                  |
| <b>SVM:</b>    | Support Vector Machine                          |
| <b>OS:</b>     | Operating System                                |
| <b>OpenCV:</b> | Open Computer Vision                            |
| <b>VNC:</b>    | Virtual Network Computing                       |
| <b>GPIO:</b>   | General-Purpose Input/output                    |
| <b>GPS:</b>    | Global Positioning System                       |
| <b>GSM:</b>    | Global System Mobile                            |
| <b>API:</b>    | Application Programming Interface               |
| <b>SIFT:</b>   | Scale-Invariant Feature Transform               |
| <b>SURF:</b>   | Speeded Up Robust Features                      |
| <b>XML:</b>    | Extensible Markup Language                      |
| <b>SD:</b>     | Secure Digital                                  |
| <b>HDMI:</b>   | High-Definition Multimedia Interface            |
| <b>BLE:</b>    | Bluetooth low energy                            |
| <b>CSI:</b>    | Camera Serial Interface                         |
| <b>DSI:</b>    | Display Serial Interface                        |
| <b>LCD:</b>    | Liquid Crystal Display                          |
| <b>WIFI:</b>   | Wireless Fidelity                               |
| <b>SSH:</b>    | Secure Shell                                    |
| <b>IDLE:</b>   | Integrated Development and Learning Environment |
| <b>GPU:</b>    | Graphics Processing Unit                        |
| <b>RAM:</b>    | Random Access Memory                            |
| <b>CPU:</b>    | Central Processing Unit                         |
| <b>HTTP:</b>   | Hypertext Transfer Protocol                     |

|              |                                               |
|--------------|-----------------------------------------------|
| <b>PC:</b>   | Personal Computer                             |
| <b>USB:</b>  | Universal Serial Bus                          |
| <b>IDLE:</b> | Integrated Development Environment            |
| <b>MP:</b>   | Mega Pixel                                    |
| <b>IP:</b>   | Internet Protocol                             |
| <b>AI:</b>   | Artificial Intelligent                        |
| <b>GNU:</b>  | GNU's Not Unix                                |
| <b>HD:</b>   | Hard Disk                                     |
| <b>ISRO:</b> | Indian Space Research Organisation            |
| <b>NASA:</b> | National Aeronautics and Space Administration |
| <b>DRDO:</b> | Defence Research and Development Organisation |
| <b>KNN:</b>  | K-Nearest Neighbor                            |

## 1. INTRODUCTION

Today, the security system field is a very advanced one. The security is the one of the most important section of person's life. Security of the house and the family is important for everybody. The new devices can run any program, in more effective way of doing different tasks such as turning on or turning off the device or making alerts via the built-in or external sensors. These objects can be defined as smart objects. Likewise, smart systems can provide Internet of Things (IoT). So, IoT makes either large or small network so to get collective intelligence via the processing of objects' knowledge [1]. Numerous options will be provided through smart objects, and each day there are various smart objects. Thus, interconnected intelligent objects can be more helpful with each other and with other objects.

The IoT can be applied in smart cities in order to give various benefits that enhance citizens [2- 3]. In other terms, smart homes can be made by utilizing the IoT. It has the ability to control and automate exact things of houses such as lights, doors, fridges, distributed multimedia, windows and irrigation systems. Some governments apply the IoT to control and care the earth, which is called smart earth, for expecting and preventing any type of human disaster such as tsunamis, earthquakes, fire or floods [4-8].

Computer vision provides face detection, face recognition and body detection for people that are very interesting in applications such as the IoT. Computer vision can present more security system in the IoT platform for smart houses. It has abilities to recognize a person in the incorrect area and at the wrong time because this person may be a bad one for the environment. In which they utilize sensors to get information and data achieved in certain conditions. Then, the camera starts capturing an image. From that point onward, a person views that image and assesses the circumstances [9].

Real-time face detection and recognition systems are very important for photography, security and surveillance applications. Generally, it recognizes and tracks persons in public areas such as houses, offices, airports, shopping centers and banks [5]. This mechanism permits secure access to the house by detecting motion controlled by the embedded system.

Surveillance plays a vital role to satisfy human's safety. Security system sensors play a very important role in this field. There are two main types of sensors: outdoor sensors which

protect the interior of the home border, and sensors that protect the indoor of the home. The most common sensors are Passive Infrared Sensor (PIR), microwave sensor, ultrasonic sensor, noise detector and photo-electric sensor. Nowadays, a very popular type is wireless based security system because of its friendly nature client and less complicated installation. The main disadvantage of utilizing wired systems is to utilize the phone line for its working. Therefore, if the phone line is cut off, its fundamental control panel will not work. But a wireless system is far away from this sort of issues. It basically can connect to the power. If they have suitable battery backup, they will work even during power outages. The wireless system gives a proficient, elegant and robust solution for the issue of remote home access, security and surveillance with human detection. Real-time human detection systems are vital for security applications [10].

In this thesis, Raspberry Pi 3 is utilized and Raspberry Pi camera is connected to it. The system will take an image when PIR sensor detects any movement. Then, computer vision applies to the captured image. Subsequently, the system sends the images to a smartphone via the Internet. In this case, IoT based Telegram application is utilized to see the activity and get the images and notifications.

The face is the most important part of human's body. So, it can reflect many emotions of a person. Long year ago, humans were using the non-living things like smart cards, plastic cards, PINS, tokens and keys for authentication, and to get grant access in restricted areas like ISRO, NASA and DRDO.

For security, biometrics, surveillance applications and face recognition systems are very important. Typically, it is attractive to detect and identify persons in general area. For instance; houses, airports, private offices, and shopping centers [11].

In recent years, face detection and recognition in the IoT have been researched and developed. Likewise, they can be used for many vision-based applications such as security applications. The face is a complicated multidimensional building and requires a superb calculating method for recognition. The most important features of the face image are nose, eyes and mouth which are related to facial extraction [12-13].

Face detection and recognition system is simpler, cheaper, more accurate and non-intrusive process as it is compared to other biometrics. The system will fall into two

categories; face detection and face recognition. There are many methods to implement face detection such as Haar-like features, Eigenface and Fisherface. Then, analyzing the geometric features of facial images, such as, distance and location amongst eyes, nose and mouth provided by several face recognition techniques [9, 14].

The final techniques utilize the AdaBoost learning to detect the face from the Haar-like features called as the Viola-Jones technique. Generally, the face detection techniques are based on the Viola-Jones techniques. This technique offers a better detection rate by comparing to another method. The computation cost of the Viola-Jones technique is relatively low. So, this face detection method becomes the most famous method. However, if the face is overlying by the other objects, the technique fails to detect the faces [15].

There are few techniques for fetching the most important features from face images to implement face recognition. One of these feature is extraction technique called Local Binary Pattern (LBP). LBP technique was produced by Ojala et al. [16]. In which describes the shape and texture of a digital image which is conceivable with LBP. This technique provides good results and efficient for real-time applications. Haar-like features and LBP are robust when compared to the others.

According to many studies [17-19] to get fast discriminatory performance and good results, LBP technique was chosen for face recognition. Also, images and shape of texture are described. This technique seems to be quite robust with different facial expressions against how the face looks in image rotation, different lightening conditions and facial aging.

Face recognition system classified into steps; face detection and face localization according to Haar feature-based Haar cascade classifier. So, with using weighted LBP algorithm, face features will be extracted [20].

LBP that is summarizing the local special structure of a face image and its working on local features using LBP operator. LBP was known as an ordered set pixels' intensity's binary comparisons between the center pixels and its eight surrounded pixels in the image. Thus, from local binary, face image has extracted its features for face recognizing. LBP uses variant face image in the database which compared with the input face image. Viewing lighting and environmental conditions have an effect on the face image [21-26].



By normalizing intensity values in neighborhood, LBP will generate the binary code that describes local texture pattern. From the LBP face image, the nose and eyes area are extracted, and for each image's pixel the LBP histograms will be drawn [12-13]. LBP features have performed very well in different applications, including texture classification, segmentation, surface inspection and image retrieval.

On the other hand, human's body is very important for security. For this purpose, a body detection algorithm is implemented to detect people for security system. Histogram of Oriented Gradients (HOG) algorithm proposed to detect the humans in the captured image. The Support Vector Machine (SVM) learning classification is used to detect the human from the HOG features. Generally, the human detection techniques are based on the Dalal and Triggs technique. This technique offers a better detection rate by comparing to the other methods. The computation cost of the Dalal-Triggs technique is relatively low. So, this human detection method becomes the most famous one [27].

### **1.1 A Literature Review**

There are some papers proposed for security system. All security applications are utilized for various purposes. In literature, there are a few studies by using a camera in IoT and sensors. In some cases, these mixtures are used to enhance the work conditions, getting more information without traveling to the exact location, or to get data about something. One of these utilizes the proposal of mixture for learning. For example, the IoT could help learning and demonstrating different information amongst master degree and undergraduate studies through gathering information and finding the best route to train. Thus, they can assist to protect the people of cities.

Creating or studying on maps, which is called Cartography, is another utilization of computer vision [28]. The most famous sample of cartography is Bing Maps and Google Maps, which are useful services for people. Computer vision is programmed for recognizing some particular sections in maps, such as buildings, roads, water or fields. This is a case of how to apply computer vision in smart earth [2]. A smart monitoring system utilizing PIR sensor, Raspberry Pi, and mobile device have been proposed. They utilize smoke detector for

detecting fire. After capturing the picture to client mail by Wi-Fi, the client will notify about the burglar or fire. They utilized smoke detection algorithm and background subtraction algorithm for movement detection [29]. A security system which has been performed and the user will notify by email or twitter if any person comes at the door. They let the proprietor control the door remotely and for preventing unauthorized access [30]. Abstraction of smart supervisor system using IoT based on embedded Linux O.S. with ARM11 architecture has been performed. They have implemented real-time video monitoring system and acquired data. In this system, they have also used PIR, temperature and humidity sensors. The system requires authentication from user to activate. If the system detects human, it will send it to the server or user's smartphone [31].

In the proposed system, the camera utilizes to achieve the image when a movement is detected through PIR sensor. Then, computer vision module will be applied to the captured images to detect and recognize the faces, then it will send it to a user's smartphone. This system is very useful and important for someone who wants to secure a place.

## **1.2 The Aim of Thesis**

In this thesis, a real-time face detection and recognition system proposed that will equip for handling images very quickly. While acquiring very high true positive face detection and recognition rate in the IoT for security applications. Also, a body detection application is proposed to detect humans in a security system.

The main objective of this thesis is to protect home, office and to identify people. For this purpose, PIR sensor uses to detect movement in the specific area. Afterwards, the Raspberry Pi camera will capture the images. Then, the face will be detected by using Haar-like feature algorithm. By applying this method, it can find faces in real-time image and video very fast. Finally, apply LBP algorithm to recognize faces. The proposed systems are real-time, quick and computation cost.

In the thesis, the Raspberry Pi single-board computer is a heart of the embedded face detection, face recognition and body detection system. It controls each of the peripherals. The

board has a 1GB of RAM, 1.2MHz ARM CPU, Wi-Fi and Bluetooth. Raspberry Pi camera utilized for image obtaining.

The publications that are published as a result of this thesis work are listed below [32-33].

- “A New IoT Combined Face Detection of People by Using Computer Vision for Security Application”, IEEE International Conference on Artificial Intelligence and Data Processing (IDAP17), September 16-17, Malatya, Turkey, 2017.
- “A New IoT Combined Body Detection of People by Using Computer Vision for Security Application”, IEEE International Conference on Computational Intelligence and Communication Networks (*CICN 2017*), September 16-17, Girne, Cyprus, 2017.

### 1.3 Organisation of the Thesis

The rest of this thesis is organized as follows:

**Chapter 2: State of art**, gives a definition of IoT, top IoT applications, IoT platforms, IoT architectures and short information about computer vision. Also, the hardware and software requirements mentioned.

**Chapter 3: The proposed methodology that utilized for face detection is explained.** After that, demonstration of all the results of face detection application.

**Chapter 4: The proposed methodology that utilized for face recognition is explained.** This chapter starts with short information about face recognition and its steps, and all the results are mentioned.

**Chapter 5: The proposed methodology that utilized for body detection is explained,** and illustrates the results of body detection.

**Chapter 6: Conclusion**, this is the final chapter. This chapter is dedicated to conclude remarks on the work of this thesis as well as providing the final insight into possible ways of further enhancing the work in future studies.

## **2. STATE OF ART**

### **2.1 Internet of Things**

Nowadays, the IoT is quickly growing technology with business circumstance. It is the confluence of wireless networks and computing. Recently, the IoT is one of the most utilized technologies with the interest for some countries. IoT connects the physical things such as buildings, vehicles and different gadgets with embedded smart sensors, and enables these things to exchange and gather information [34-37].

The IoT is becoming popular in many sides of life, such as smart security, smart cities, healthcare, smart transportation, smart grids and online business. The objectivity of utilizing IoT is to share information and knowledge with everyone in everywhere around the world, from food to PCs and to automate various procedures to enhance daily life [38]. The IoT is a dynamic worldwide network of daily things connecting to the Internet. It is the interconnection of virtual and physical objects accessed through the Internet. These things can be objects of the physical world or data of the virtual world [37]. The IoT is the name of the Internet of objects, belongs to a wireless network among objects. The IoT is making smart world, which is the fusion of different and ubiquitous things, smart homes, and smart cities, with all gadgets fit for cooperating amongst themselves [2].

Wireless Sensor Networks are the most essential, which are the center of the IoT. A Wireless Sensor Network interconnects sensors, so as to get information with a server or unique system to work and automate tasks in one place [35]. The best definition for IoT is an interconnection of different and ubiquitous objects among themselves via the Internet. The objective of the IoT is to interconnect the entire world via the making of various smart places to automate, enhance and facilitate day to day life [39]. IoT is nowadays becoming a part of each side of human's life.

### 2.1.1 Top IoT Applications

It is true to propose that IoT is nowadays become a part of each side of person's life. Some IoT is announced by a new company every day enabled product. These are the most important IoT applications:

- **Home Automation**

Persons monthly utility bills will reduce by using connected home, and advise us when to irrigate plants. In home automation, more companies are energetic than other applications in IoT [40].

- **Smart City**

Traffic congestion problems are solved through IoT solutions in smart city applications, and create cities safer than ever before and reducing noise and pollution. People can light their streets more efficiently through the smart city [38].

- **Wearable**

Wearable remains a hotly topic in IoT. Every IoT startups, wearable's creator Jawbone is likely the one with the greatest financing to date. There are many wearable objects such as smartwatch and Sony smart trainer [34].

- **Smart Grid**

The smart grid is one of the particular IoT applications. A future smart grid guarantees to utilize data about the conducts of power suppliers and purchasers in an automated manner to enhance the reliability, effectiveness and economics of power [41].

- **Health Care**

Peoples health will monitor by sensor devices, track day by day our activity or remotely monitor an old age family relative. Health care system and smart medical gadgets not only utilize for companies, will also utilize for the good health of people [37].

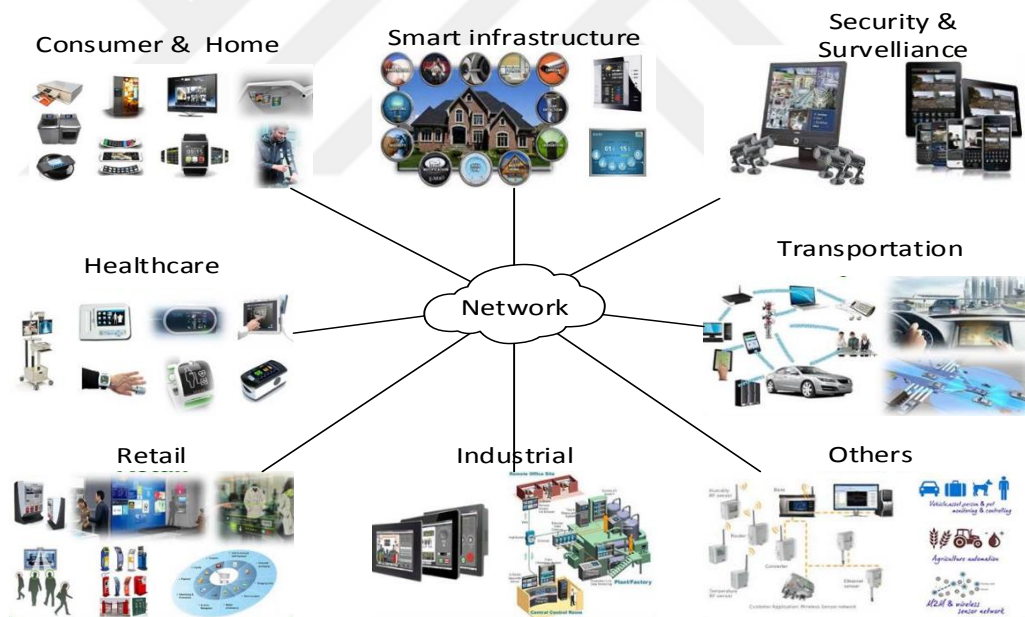
- **Transport and Logistics**

The first business sectors interested in IoT applications is Transport and Logistics. Transport and Logistic applications integrate with GPS, GSM, and light sensors [37].

- And IoT can be used in another field such as:

- Security
- Agriculture (smart farming)
- Smart retail (Intelligent shopping)
- Traffic monitoring (Google traffic)
- Food services
- Hospitality
- Banks

The application areas of IoT are given in Figure 2.1.



**Figure 2.1.** The application areas of IoT

In Figure 2.1, IoT has been used in many application areas. Some of these IoT platforms have attributes that others don't have. Be that as it may, none of them has a module of computer vision that permits working with images as sensors and images with sensors.

### 2.1.2 IoT Platform

As explained before, we have to interconnect them among themselves to get the best potential of smart things. Although, require a brain which can notify and manage the smart things and sometimes to work as the brain for some different things such engine. This brain is an IoT platform. There are various IoT platforms with various benefits and disadvantages [42]. Have numerous IoT platforms for example, Carriots [43], ThingWorx [44], Nimbits [45], ThingSpeak [46], Kaa [47], and Telebot API [48].

Some of these IoT platforms have attributes that others don't have. It may none of them has a module of computer vision that permits working with images as sensors and images with sensors. A camera can be utilized as an engine, namely, this camera can be interfaced and take images under specific conditions. The target is to utilize the cameras' images with a sensor. For instance, the camera could be connecting and send the images when a specific condition was an achievement. The proposed system is one possible solution to utilize the camera with a sensor utilizing computer vision to detect a specified thing in images [5].

### 2.1.3 IoT Architecture

IoT architectures consist of various appropriate of technologies. It helps to see how different types of technologies link to everyone and to communicate the adaptability, seclusion, and setup of IoT arrangements in various cases [49]. The task of all layers is portrayed beneath.

- **Application Layer**

Let the user to interact and supply the user experience for the systems.

- **Management Services Layer**

In this layer, some process of information will have done via security controls, analytics, management in devices, and process modeling.

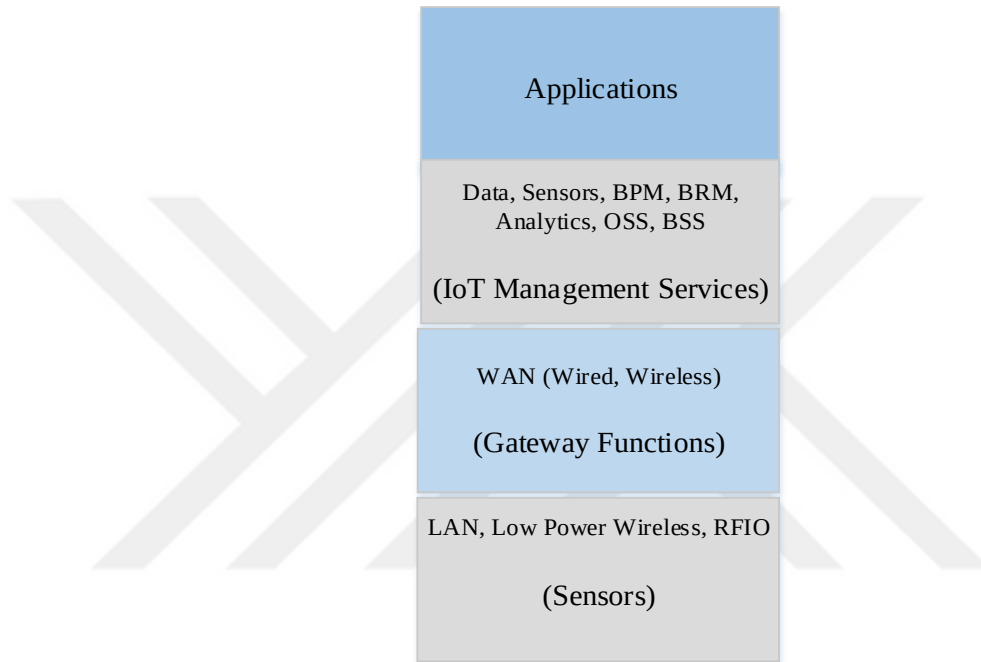
- **Gateway Layer**

This layer is able to strong and high execution wired or wireless network infrastructure as a vehicle medium.

## - Sensor Connectivity and Network

This layer is capable of interconnecting of the digital and physical worlds letting real-time data to be gathered and processed.

The IoT architecture is given in Figure 2.2.



**Figure 2.2.** The IoT architecture

## 2.2 Computer Vision

The only thing that computer can do is processing zeros and ones. However, the purpose of bearing Artificial Intelligence (AI) is to serve the possibility of making programmers that let computers to learn. In 1955, John McCarthy instituted this word in the conference of Dartmouth. Inside the AI, one of the fields is that the computer vision.

Computer vision is a perplexing and important field that focuses on how computers have the ability to make to earn high-level understanding from digital images or videos, for recognizing and understanding digital images or the characteristics of images, and mainly works on the extraction of high dimensional information from this present reality for producing numerical or typical data. This will allow to identifying humans, objects, animals,



or a position in images. In this way, the objective of the computer vision is that a machine is capable to understand the world [50].

To reach this goal, if someone want to recognize and process something in a picture, can be find a lot of algorithms for that purpose. Some algorithms use for obtaining the components of the dataset that can be utilized to train the model are Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), HOG–LBP, Scale-Invariant Feature Transform (SIFT), and Speeded-Up Robust Features (SURF) [51-56], but others are the necessities to prepare the model utilizing the extricated features that got it before. Logistic Regression and Support Vector Machine (SVM) are two examples of these algorithms [57-58]. Nevertheless, to get a good model, it need difficult task. User have to take a lot of good images and try many times with another gathering of desperate images to check that model functions well. As well as, for solving problem creating a model is important, since the utilization of general models could make the accuracy less. Instances of these models could be the example models of a few applications such as OpenCV library. In this study, this kind of models will be utilized. In the proposed system, computer vision uses to detect and recognize the faces in the images.

## **2.3 Hardware Implementation**

The next hardware components are used in this thesis as following:

### **2.3.1 Raspberry Pi 3**

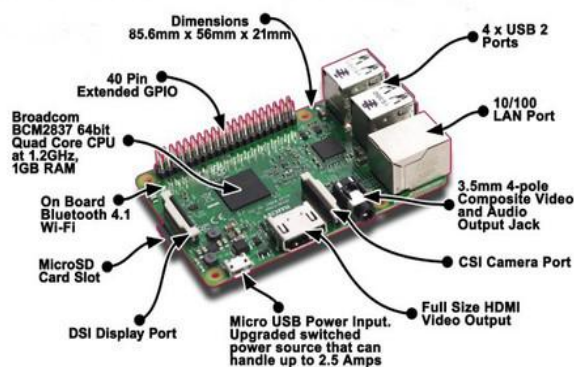
The Raspberry Pi 3 used in the proposed system. Raspberry Pi is a small credit-card sized computer that has ability for many functionalities execution such as in security systems, real-time application and military applications. The different parts of Raspberry Pi 3 are shown in Figure 2.3.

For installing OS/booting/long term storage the Raspberry Pi has SD card slot. And 32GB would be the total memory of the SD card. And Broadcom BCM2836 processor has also applied to the Raspberry Pi 3. Raspberry Pi 3 runs at frequency of 1.2GHz with memory capacity to 1Gigabyte and BCM2836 is high powered ARM Cortex-A53 primarily based 64-bit quad-core processor. It has Micro USB Power Port provide 700mA at 5A and RCA Video

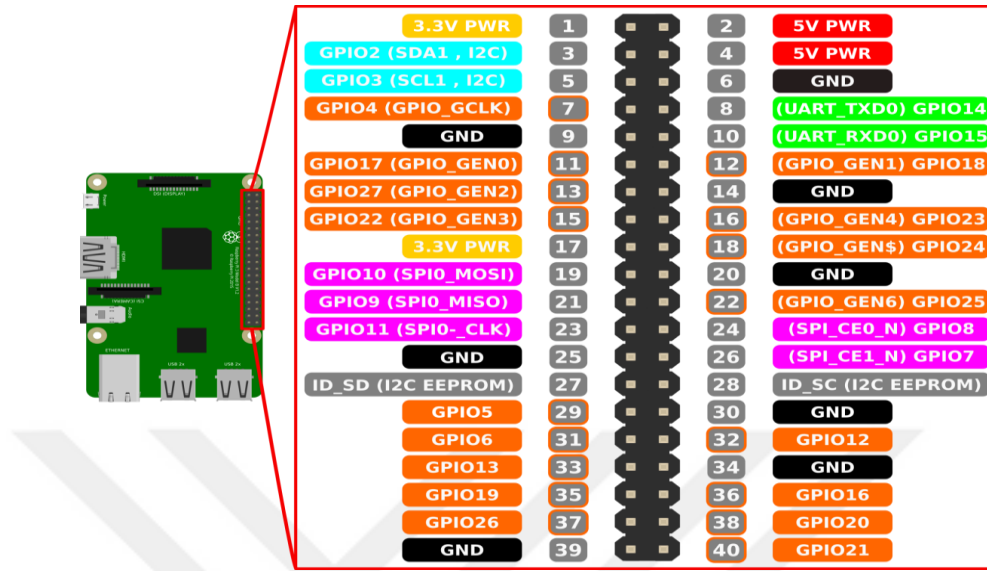
Out is connected to show if HDMI output is not utilized. It is mainly utilized to carry audio and video signals. They are called as A/V jacks. Using HDMI to acquire stereo audio will lead to obtain Out Digital audio. Here analogue RCA connection is utilized. That has forty pin GPIO Header for connecting the external gadgets for communicating with the processor. In Figure 2.4, the input and output pins of the GPIO are mentioned, demonstrates the present status of the appliances. 8 GPIO pins for external fringe connections supported by Raspberry Pi3. 2- 3.3V, 2-5V pins for power supply connected to the external gadgets. With the Python is a default programming language for the Raspberry Pi with support of Java, C, C++, Ruby and Perl. The Raspberry Pi 3 has features Bluetooth Classic and Bluetooth Low Energy (BLE), and 802.11 b/g/n 2.4 GHz Wireless LAN. A quicker connection can be picked without external gadgets assistance.

Furthermore, for the Internet connection the Raspberry Pi 3 likewise has Ethernet port to be connected on the web, additionally to connect pi camera there is camera interface (CSI), display interface (DSI) is utilized to connect LCD screen, full HDMI port is utilized with HDTVs and screens with HDMI input and 4 USB 2.0 ports is utilized to connect keyboard or mouse. The Raspberry Pi 3 dimension is around 85.6mm x 56mm x 21mm [59]. Many advantages can be discovered in the Raspberry Pi such as high security, effective detection, automatic based, no missing of data, low cost or quick alert to owner.

This Raspberry Pi 3 works on the Windows 10 IoT and all the recent ARM GNU/Linux distributions. Raspberry Pi 3 contains an OpenCV based image processing library. And this library can be utilized to perform the operations on the pictures while performing face detection.



**Figure 2.3.** Raspberry Pi 3 block diagram

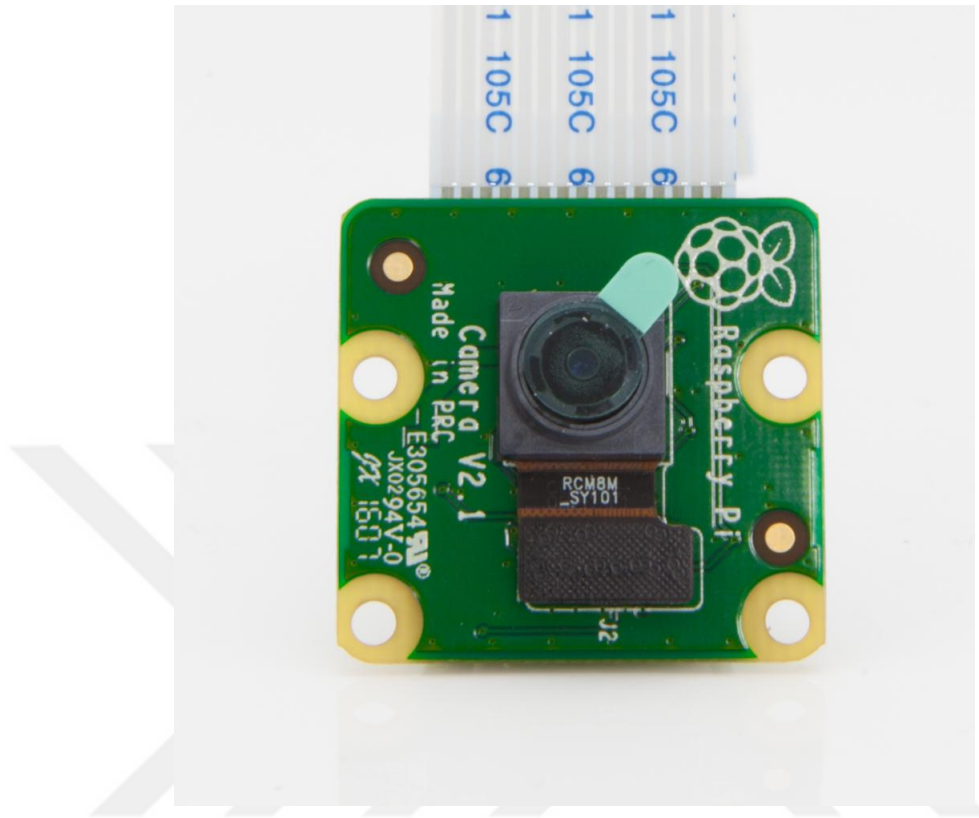


**Figure 2.4.** GPIO pins of Raspberry Pi 3

### 2.3.2 Raspberry Pi Camera

For taking full HD 1080p photo and recording video in full HD that can be controlled programmatically later, the camera module would be such a good option for the Raspberry Pi.

The camera can be connected to the Raspberry Pi 3 by small printed circuit board. Raspberry Pi camera board directly connected with the CSI connector that exists on the Raspberry Pi 3. The Raspberry Pi camera module attaches to Raspberry Pi via 15 pin Ribbon cable to the committed 15-pin MIPI Camera Serial Interface (CSI) that designed specially to interfacing to cameras. Outrageous high information rates would be the responsibility of the CSI bus, and it conveys pixel information to the BCM2835 processor [60]. A small 8MP pi camera can be noticed that is similar to the ones utilized on smartphones. For enabling camera must go to the Raspberry Pi configuration settings and selecting enable camera to begin utilizing the camera. Raspberry Pi camera V2.1 is used in the proposed system. Figure 2.5, shows Raspberry Pi camera board.

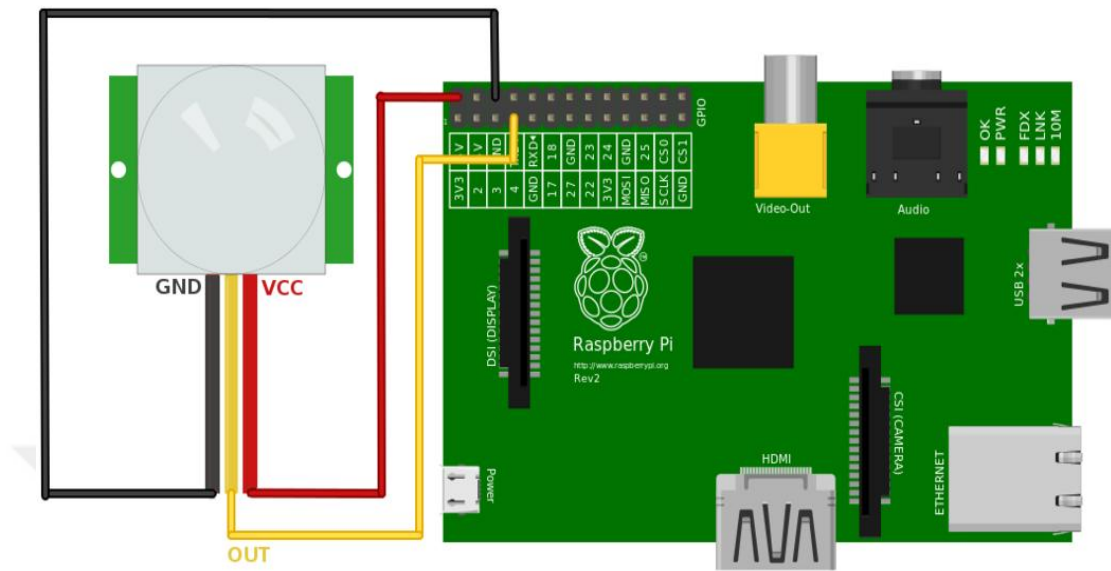


**Figure 2.5.** Raspberry Pi camera

### **2.3.3 PIR Sensor**

A passive infrared sensor is an electronic sensor that utilized to measure infrared (IR) light radiating from items in its field of view. It can be utilized for movement detecting by receiving infrared radiation in surveillance area [29].

To sense motion, PIR sensors will give permission. They are small, easy to use, inexpensive, pretty rugged, low power, easy to interface with, and it has a wide lens range. In this manner, they are usually utilized in businesses or homes. In the proposed system, PIR sensors will connect with the Raspberry Pi. The PIR acts as a digital output. To detect motions, connect ground to ground and power the PIR with 5V. Then, connect the output to a digital pin as shown in Figure 2.6.



**Figure 2.6.** Connecting PIR sensor to Raspberry Pi

The primary function of the project is the movement detection. When motion detects, the client can be customizing the activity. For example, a script was as an option to test the project that execute on the Raspberry Pi to send a message notification and captured images. The end client received a message and he could do the needful whenever the motion detects, for example, see the movement and notification.

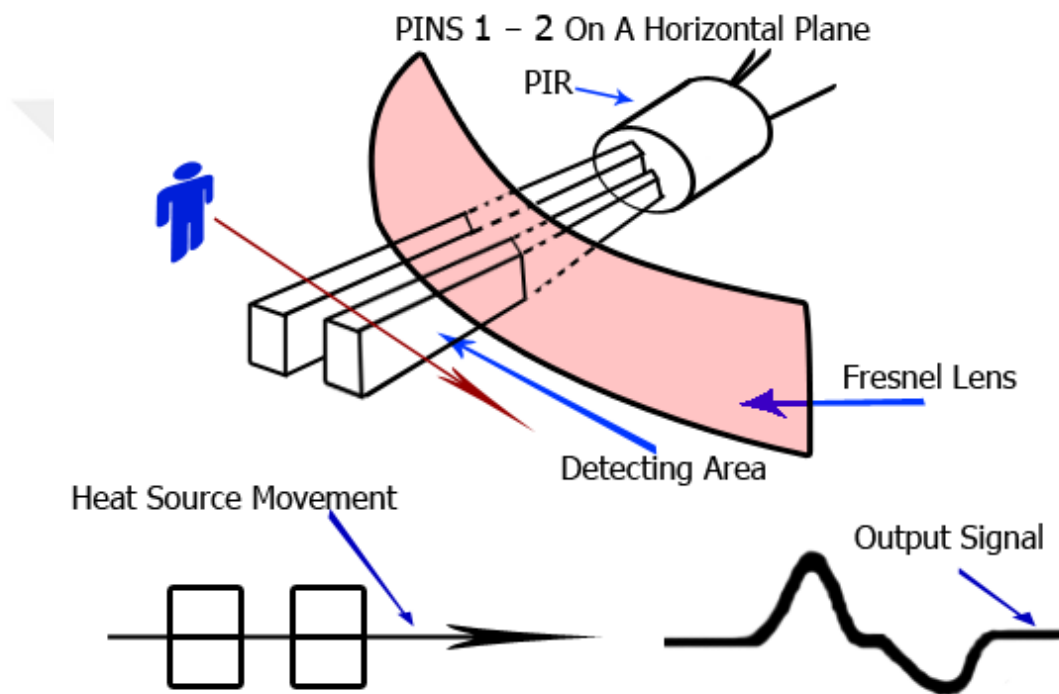
The linchpin of the security system is the movement detector, since it is the main gadget that detects when somebody is in home when they should not be. The movement detector utilizes one or numerous technologies to detect movement in a zone. If a sensor is bag down, a signal is sent to security system's control board, which is connects to the monitoring focus, notifying and the monitoring focus to a potential danger in home.

The security system can utilize two types of movement detectors: microwave movement detectors and passive infrared (PIR) detectors. PIR detectors are the most generally found movement detector [61].

PIR sensors allow to sense motion as shown in Figure 2.7. When the person moves in or out of the sensor's range, the PIR sensors utilize to detect weather. It needs to detect when a person has entered or left the room or the house, or has approached to turn on or off the lights,

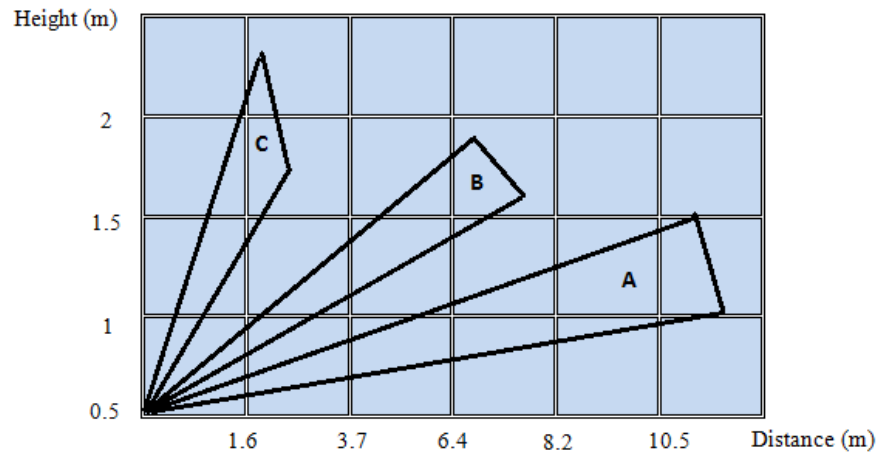
electrical fan or security alarm, PIR sensor sends signal to the microprocessor. Which has been utilized it in the proposed system to security alarm.

The surrounding heat radiation is grouped by a portion lens and guided to the pyro-sensor. When a heat source moves in the detection zone, the heat radiation changes and the pyro sensor discharges a voltage. This voltage is surveyed by the downstream electronics.



**Figure 2.7.** Sense motion by PIR sensor

The detection lens is split into different parts to cover much bigger area and all part has Fresnel lens. As shown in Figure 2.8, here the area of A zone is conceivable detection of thing less than 12 meters in distance and 1.5 meters high. The area of B zone is detection of thing less than 6meters in distance and 1.5 meters high. Lastly, C zone is one with less than 3 meters' distance and 2.5 meters high. Fresnel lens compresses the light, providing a bigger range of IR to the sensor and it will extension over some tens of degree width [62].



**Figure 2.8.** Possible detection range

### 2.3.4 Smartphone

Smartphone is a cell phone that performs many of the functions of a computer; usually it has a touch screen interface, an operating system capable of running downloaded apps and Internet access. Numerous things should be done via smartphone as much as a personal computer able to do, and because of its mobility much more. Thus, screen size does matter, because the higher resolutions produce seeing pleasurable.

The interaction amongst system and client can be done by smartphone. Smartphone has incredible power in calculation as well as a very convenient operation such as wireless Internet access by Wi-Fi or camera monitoring. The system becomes smarter and more intelligent with the assistance via powerful smartphone applications [59].

Also, IoT is an ongoing development of the Internet by which everyday things have communication capabilities which permit them to send and get data. To see the activity and get notifications when movement is detected, IoT based application can be utilized remotely. To get images and notifications, telegram application utilized that has already installed on the smartphone. In the proposed system, to see the captured images remotely, smartphone is utilized and furthermore, it can get the notification messages at the same time.

### **2.3.5 SD Card Memory**

The design counts on SD card instead of built in hard disk or strong state drive for booting and long term storage. An SD Card (Secure Digital Card) is an ultra-flash memory card and intended to give high capacity memory in a small size. This board intend to run Linux kernel based operating systems. In the proposed system, 16GB SD card used to install Raspbian operating system on Raspberry Pi.

### **2.3.6 Power Supply**

A very simple power supply can be seeing on the Raspberry Pi. It utilizes a Micro USB connection to power itself. The power source utilized for the gadget is a 5200mAh external battery for tablets and smartphones.

## **2.4 Software Implementation**

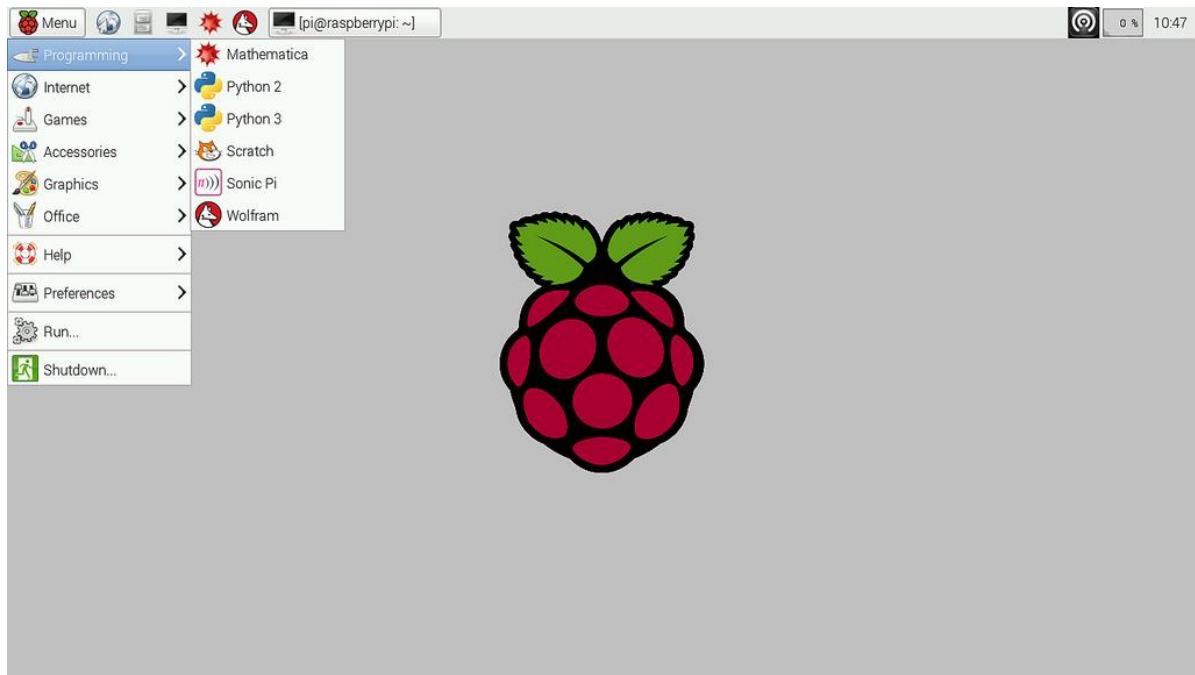
The next software is used to implement in this thesis:

### **2.4.1 Raspbian OS**

Raspbian based on Debian optimized for the Raspberry Pi hardware is an open source OS, is the current prescribed system, released in July 2012. The GPU hardware can be gotten via a firmware image that has been already loaded into the GPU at boot time from the SD card. The Raspberry Pi works on the basis of Raspbian operating system.

In order to install Raspbian operating system, next out of box software (NOOBS) which has to be installed. The initial step to install operating system, the drive must be allocated and SD card adaptor can likewise be utilized for this reason. To write the disk image, utilize WINDISK 32 utility. It accessible from source forge project which a compress file. Then, extract and run the file. Subsequently, select the image file and click write to begin the process and wait for writing process to finish. At last, close the image and eject the SD card [29, 59]. Figure 2.9, demonstrates the Raspbian operating system installed.



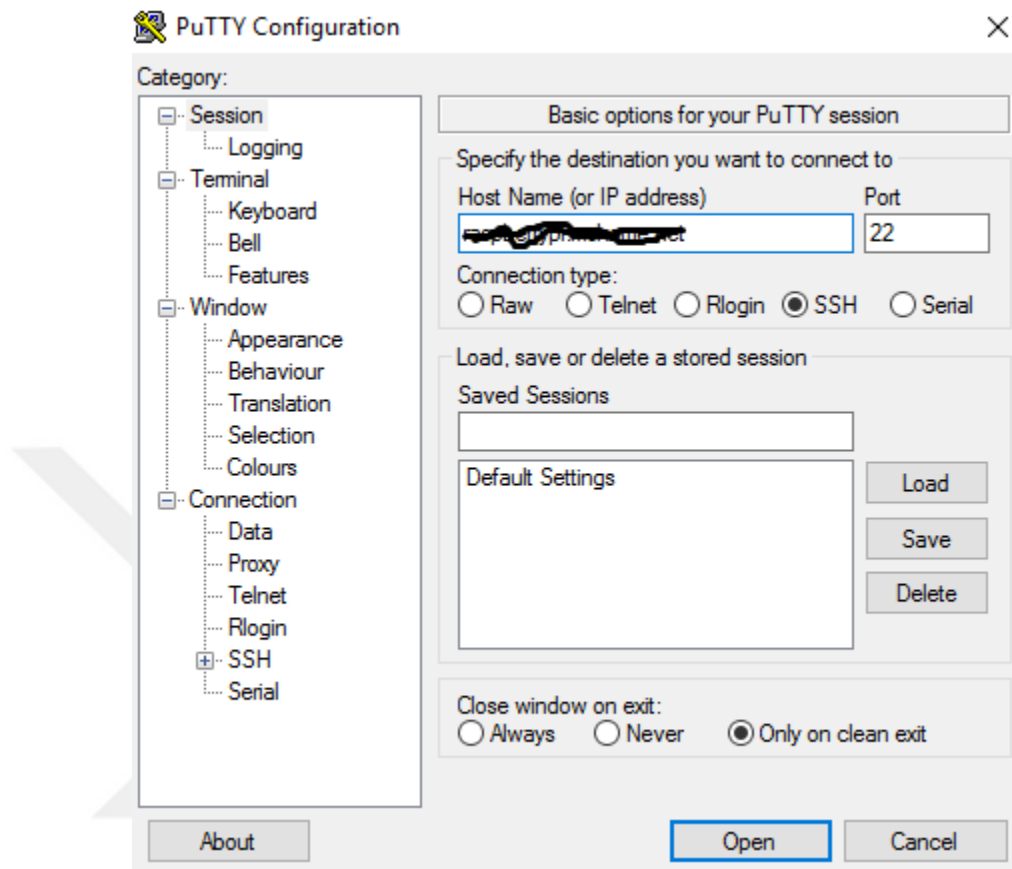


**Figure 2.9.** Installed Raspbian OS

## 2.4.2 PuTTY Configuration

PuTTY is a free and open-source terminal emulator; serial console and network file transfer application. A few network protocols can be supported, including SSH, Telnet, rlogin, SCP, and raw socket connection. It can likewise connect to a serial port. Raspberry Pi can work in standalone mode and programmed by utilizing PuTTY. Via hostnames or IP address of Raspberry Pi, it can access command window of Pi after installing PuTTY on the PC [31].

On windows, to get the Raspbian desktop, needs to download an SSH client. The most common is called PuTTY. PuTTY does not have an installer package. It is only a standalone .exe file. After, to run the PuTTY, need to type the hostname or IP address of the Raspberry Pi into the hostname field with select SSH in the connection type field as appeared in Figure 2.10. Then, click the open button.



**Figure 2.10.** PuTTY configurations

### 2.4.3 VNC Server

Virtual Network Computing (VNC) is a graphical desktop distributing system that utilizes the remote frame buffer protocol to remotely control another PC. Through VNC server transferring the graphical screen updates back in the other direction and transmits the mouse and keyboard events from one PC then onto the next above a network. Popular utilizes for this technology include remote technical support and getting to files on one's work PC from one's home PC, or vice versa [59].

These steps to install VNC server:

#### **Step 1: Installing VNC server**

To install VNC server on Raspberry Pi go to update the operating system to the newest version with the command:

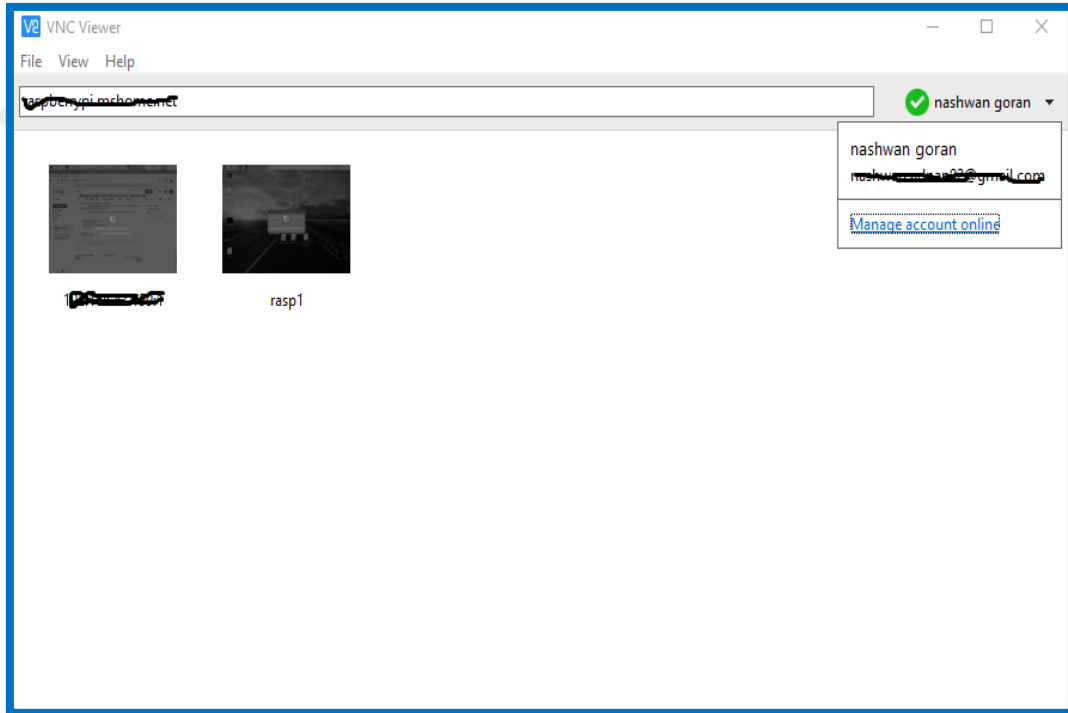
```
$ sudo apt-get update
```

After updating finish, needs to install VNC server with command:

```
$ sudo apt-get install tightvncserver
```

### **Step 2: User Side (PC)**

Downloading of VNC user program and dynamically enters hostnames or IP address of Raspberry Pi to work pi via computer. Installed VNC server shown in Figure 2.11.



**Figure 2.11.** Installed VNC servers

### **2.4.4 Python Language**

Python is a superb and generally utilized intense high-level programming language that's easy to utilize and with Raspberry Pi can be connected projects to the real world. Python syntax is clean, with an accentuation on decipherability and utilizes standard English keywords. Begin via opening IDLE from the desktop. Python is the official programming language of the Raspberry Pi. Raspbian OS comes advanced with python. Python programming language utilized in the proposed system to detect movement, capture image, face detection and send images and notifications to smartphones.

To install python library on the Raspberry Pi, type the command underneath:

```
$ sudo apt-get install python-dev
```

```
$ sudo apt-get install python-pip
```

```
$ sudo apt-get install python-rpi.gpio
```

### **2.4.5 OpenCV Library**

OpenCV is a library with open source image processing for programming functions that aimed real-time computer vision. OpenCV has more than 2500 optimized algorithms which are existed in the library. It includes a machine learning algorithms and comprehensive set of both classic and state of the art computer vision. These algorithms can be utilized for detecting and recognizing faces, track camera movements, identify objects, track moving objects, classify human actions in videos, produce 3D point clouds from stereo cameras, extract 3D models of objects, for generating a high-resolution image of an entire, the scene stitch images will use together, removing red eyes from images which caused through camera flash, discover similarity between images and an image in the database, recognize scenery and establish markers to overlay it with augmented reality and follow eye movements [63].

Under the open-source, the library is cross-platform and free for utilize BSD license. OpenCV accessible for Python, Java, C, C++ and MATLAB interfaces and supports Windows, Linux, Mac OS and Android. In the proposed system, OpenCV library used to detect and recognize the faces from the captured image by utilizing Haar-like feature algorithm.

### **2.4.6 Telegram Bot API**

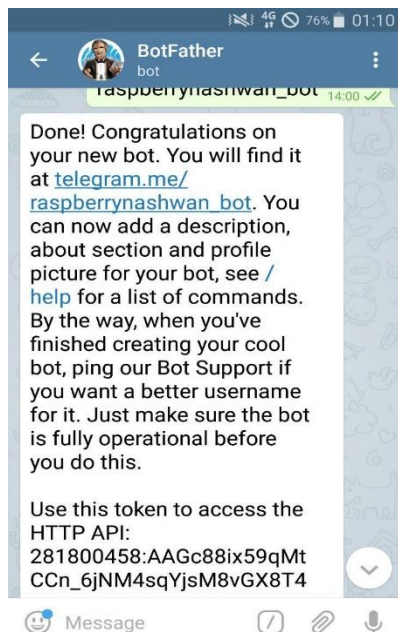
The Telegram Bot API is an HTTP-based interface made for developer's keen on building bots for Telegram. Bots are third-party applications that running inside Telegram. A bot may be turned into a smart newspaper. Any published relevant content will send to the client instantly it's possible. Furthermore, by external services content a bot can enhance Telegram chats easily such as Image Bot, Gif Bot and Music Bot. Likewise, bot provides users with

translations, alerts, formatting, weather forecasts or other utilities. TeleBots are special accounts at the core that for setting up no telephone number has required ever.

`/Newbot` command can be utilized for creating a new bot. username and name will be asked by the Bot Father, then a mandate token for new bot will be created. For contact details, the name of the bot is displayed. The username is a short name; to be utilized as a part of notice and the username is saved in the database inside the Telegram application.

The token is a combination of capital and small letters, and numbers for example token maybe generated like this: 122001443: ABHdqTcdCh1vGWJxfWesfSAs0K4FQLD. It is needed to approve the bot and send demands to the TeleBot API. Figure 2.12, demonstrate the bot has been made and BotFather reveals a token for user [48].

For utilizing a Telegram Bot API with a Raspberry Pi only requires installing the Python TeleBot API library. For doing this, the command `pip install python-telegram-bot` which can be utilized from the shell. After this library has installed, it can be utilized in the proposed scripts. Telegram bot APIs are being utilized in the proposed system are only for the message could be sent from Raspberry Pi to the user's smartphone.



(a) Created newbot

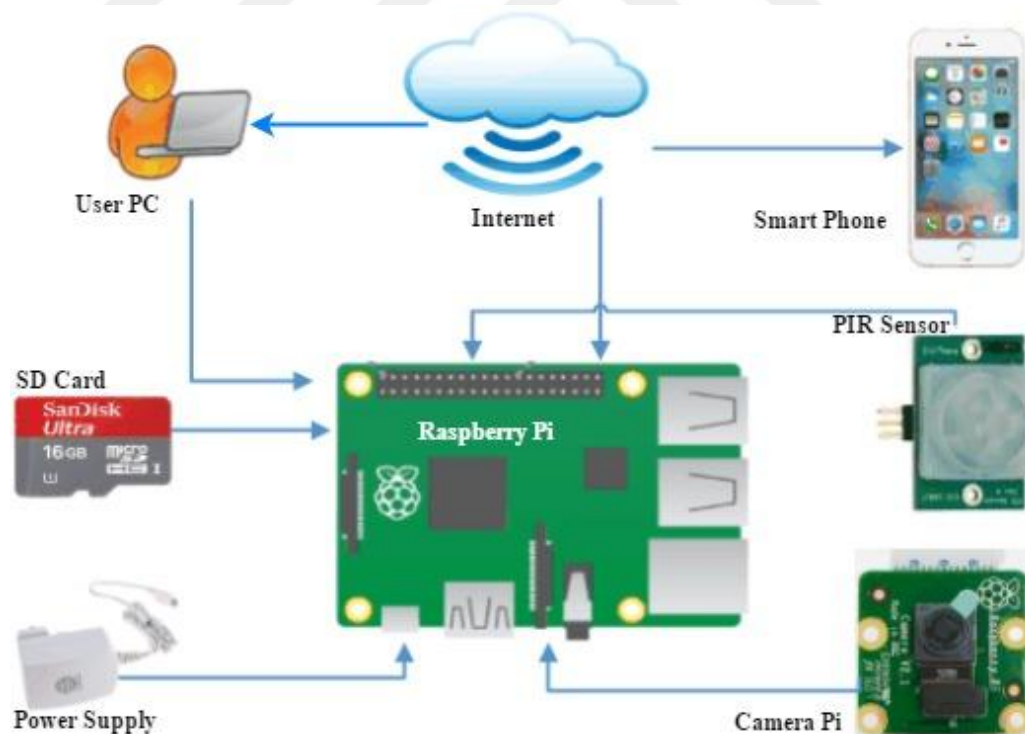


(b) Display token

**Figure 2.12.** Created boot and display tokens

## 2.5 System Design

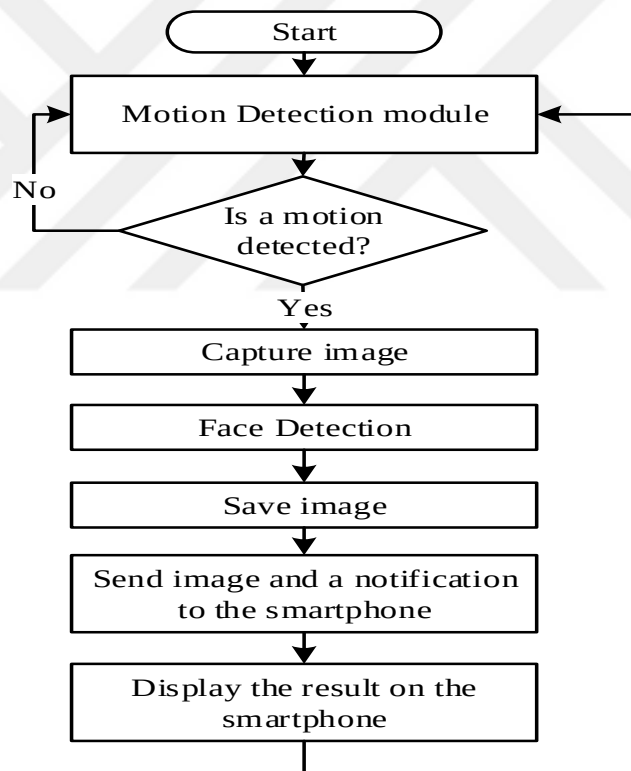
A Raspberry Pi 3 is a microcontroller which has four USB modules. Raspberry Pi camera and PIR sensor are connected to the Raspberry Pi. Raspberry Pi is connected with the smartphones through the Internet. The central platform for the image processing and signal alerting can be utilized as a Raspberry Pi board. The images are captured through a pi camera. The system has the ability to detect movement of an object through the PIR sensor. When the moving object is detected, the decision signal passes into the embedded board GPIO port. It is a general purpose input and output. Raspberry Pi camera will be turned on through the python script for capturing images and the system can check whether a face is detected or not. If the face is detected or not, it will save the images on the local drive and send them to the smartphones. Captured images and notifications on the smartphone can be seen by the client via the Internet. Figure 2.13 shows the embedded system blocked diagram.



**Figure 2.13.** System design

### 3. A NEW IOT BASED FACE DETECTION BY USING HAAR-LIKE FEATURES

In the proposed system, the camera is used to fulfil the pictures when a motion detects via PIR sensor. Then, computer vision module will apply to the captured pictures to discover the faces. After that, it will be mailed to a smartphone. This system is extremely helpful and vital to protect an area. The application can be divided into two parts, motion detection and face detection. The system will not detect the face unless there is a motion discovered. But, if a movement is detected, then the detected movement will be processed by the algorithm of face detection. Figure 3.1, demonstrates the flow chart for the proposed scheme.



**Figure 3.1.** Flowchart for proposed system

The main function of the proposed system is the detection of a movement. When a movement is detected, the Raspberry Pi sends a notification message and captured images to the smartphone. The security system can use two types of movement detectors: microwave movement detectors and passive infrared (PIR) detectors. PIR motion sensors are the main

common motion detector [61-62]. When an object moves in the sensor's range, the PIR sensors are utilized to detect it. This is required to detect a person when enters somewhere. If a motion is discovered, the PIR sensor sends a signal to the microprocessor.

### **3.1 The Proposed Method for Face Detection**

Face detection is used in different applications. Facial features are detected and others are ignored. Haar-like feature detection is the most popular algorithm for face detection. It is utilized to detect single or multiple faces. OpenCV library is used to reveal faces as they are the best open source library obtainable for image processing and they are not too much complex. In this system, the face detection part is executed in python by utilizing OpenCV library. The detection of human's face focuses on face detection algorithm. The Haar cascade algorithm looks for specified Haar-like features of a human face. The Haar-like face detection algorithm permits the face candidate to pass to the next phase of detection when any of these features is discovered. A face candidate is a rectangular part of the original image known as a sub-window. Generally, the size of sub-window is about  $24 \times 24$  pixels. For getting a set of various sized faces, the sub-window is often rotated in an order. The Haar-like feature algorithm scans the entire image with this window and denotes all respective part as a face candidate [64-65]. In the proposed method, when a movement is detected, the camera captures an image. Then, the captured image is converted to grayscale to improve speed and detection rate. By using Haar Classifier, the converted image will be processed. The program considered it as the non-face when a shoulder and head feature is not detected. Then the program will send the captured image with non-faces as notification to the user's smartphone.

The face detection algorithm has mainly four phases: integral image calculation, Haar-like features, AdaBoost training and cascade classifier. The initial step of the Haar-like algorithm is to transform the input image into an integral image. The layout of the pixel values in the original images determines the meaning of the integral image. The integral image is utilized to detect quick feature detection.



### 3.1.1 Integral Image

The integral image is an array that contains the addition pixel values of the original image. In the integral images, the value at any position  $(x, y)$  is the total of the picture's pixels over and to the left of position  $(x, y)$ . This procedure is shown in Figure 3.2. The shaded zone appears the total of the pixels over to place  $(x, y)$  of the picture. It demonstrates a  $3 \times 3$  picture and its integral image portrayal.

```
for i=1 to w
    s=0;
    for j=1 to h
        s=s+I(i,j);
        if i==1
            II(i,j)=s;
        else
            II(i,j)=II(i-1,j)+s;
        endif
    endfor
endfor
```

Original image (I)

|   |   |   |   |   |
|---|---|---|---|---|
| 5 | 2 | 3 | 4 | 1 |
| 1 | 5 | 4 | 2 | 3 |
| 2 | 2 | 1 | 3 | 4 |
| 3 | 5 | 6 | 4 | 5 |
| 4 | 1 | 3 | 2 | 6 |

The sum of pixels values in a rectangle =  $5+4+2+2+1+3=17$

Integral image (II)

|    |    |    |    |    |
|----|----|----|----|----|
| 5  | 7  | 10 | 14 | 15 |
| 6  | 13 | 20 | 26 | 30 |
| 8  | 17 | 25 | 34 | 42 |
| 11 | 25 | 39 | 52 | 65 |
| 15 | 30 | 47 | 62 | 81 |

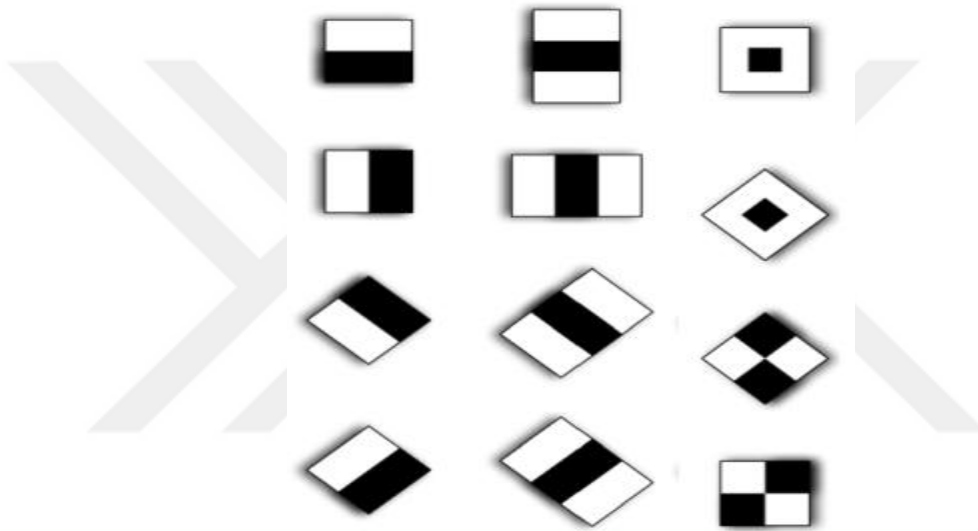
The sum of pixels values in a rectangle =  $34-14-8+5=17$

**Figure 3.2.** The pseudo code of integral image calculation and a numeric example

In Figure 3.2, the sum of pixels within rectangle can be computed with four values in the integral image. The sum of pixel values in a rectangular area can be computed by using four array references as shown in Figure 3.2.

### 3.1.2 Haar-Like Features

To detect variation in the black and white segment of the images, Haar-like features are used. This calculation is similar to a singular rectangle around the detected face. The number of possible rectangular features is 180,000 for a 24x24 detection zone. Face candidates are searched and scanned for Haar-like features in this phase. Figure 3.3, illustrates the various kinds of features.



**Figure 3.3.** Different kinds of features

Haar features are being characterized through partitioning the human face into various regions. The facial region is divided according to the size and brightness variation, and computed like:

$$\text{Value} = \Sigma (\text{pixels in black zone}) - \Sigma (\text{pixels in white zone})$$

Every feature results in a single value computed through subtracting the addition of the white rectangle from the addition of the black rectangle. The example of two features of Haar is given in Figure 3.4.



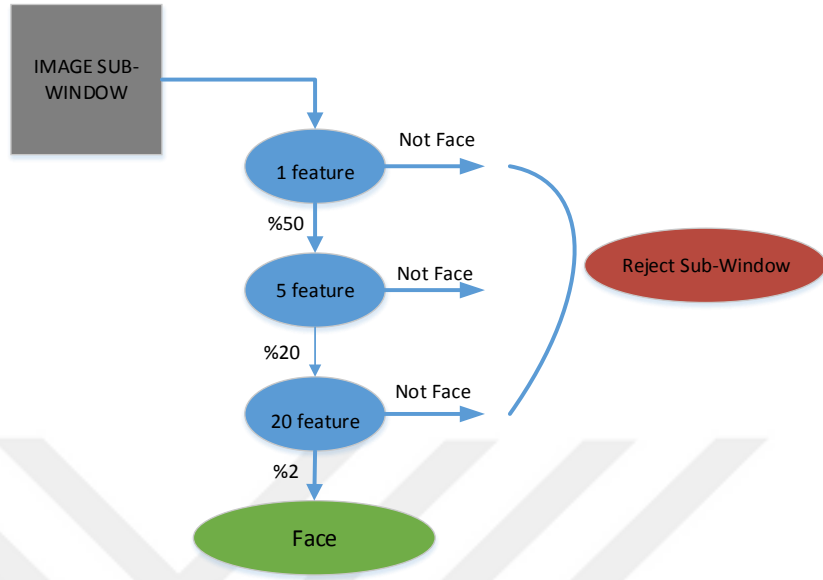
**Figure 3.4.** Haar-like feature example

### 3.1.3 Adaboost Training

AdaBoost is an algorithm in machine learning that is resourceful of building a solid classifier via a weighted group of powerless classifiers. It utilizes an important idea of Bagging, which is a technique for joining various classifiers built using similar information set. AdaBoost algorithm gives a desired area of the things repercussion not necessary background. The constancy of the better feature, threshold and polarity are important sections of the AdaBoost algorithm. The better performance feature is selected based on the weighted error it generates. The weighted error is a function of the weights belonging to the training instances. The weight of a correctly classified case is decreased and the weight of a misclassified case is stayed fixed [66-67]. AdaBoost learning procedure is quick and gives a number of desired information.

### 3.1.4 Cascade Classifier

Cascade classifier phase is the final steps for Viola-Jones face detection algorithm. Cascade stage is utilized to eliminate face candidates speedily. A cascade classifier comprises many stages of filters, deciding if a given sub window is definitely not a face or possibly a face is the work of every phase. When passing all phases or failing any phase, the candidates exit the cascade. The cascade classifier will directly reject the area as a face when the input area fails to pass the threshold of a phase. If a candidate passes all phases, the face will be detected. This procedure is demonstrated with three phases in Figure 3.5.



**Figure 3.5.** Example for cascaded of phases

Through selecting the false-alarm rate and desired hit-rate at all phases and some phases, it is possible to accomplish great detection execution. The detection rate of the full cascade is given through:

$$D = \prod_{t=1}^T d_t \quad (3.1)$$

Wherever  $d_t$  is the detection rate of the  $t$ -th phase and  $T$  is the number of phases. The false positive rate of the full cascade is given through:

$$F = \prod_{t=1}^T f_t \quad (3.2)$$

Wherever  $f_t$  is the false alarm rate of the  $t$ -th phase and  $T$  is the number of phases [64, 68].

### 3.2 Experimental Results

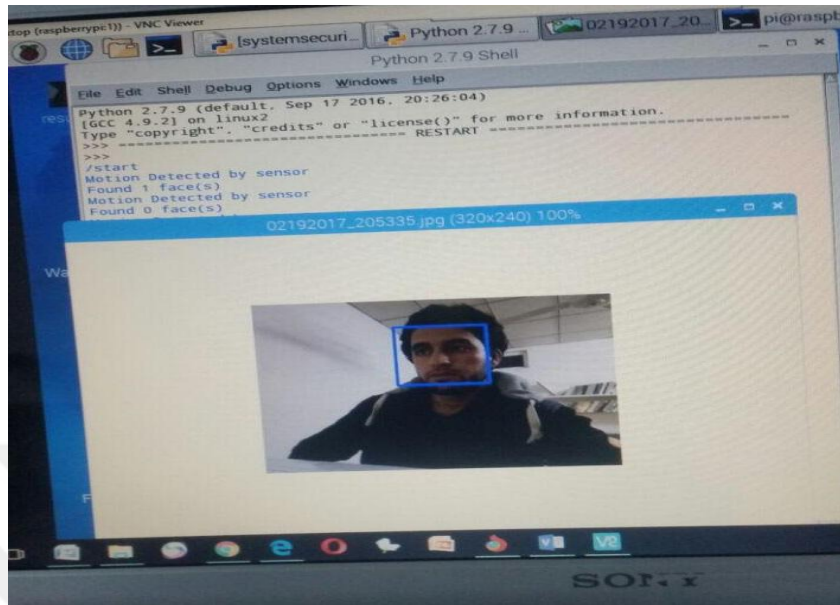
Smartphone application incorporated with the proposed system to improve a smart motion detecting camera security system and find human faces for houses and offices. The experimental setup of the proposed scheme is shown in Figure 3.6, other than the hardware configuration the software's necessary should also be installed.



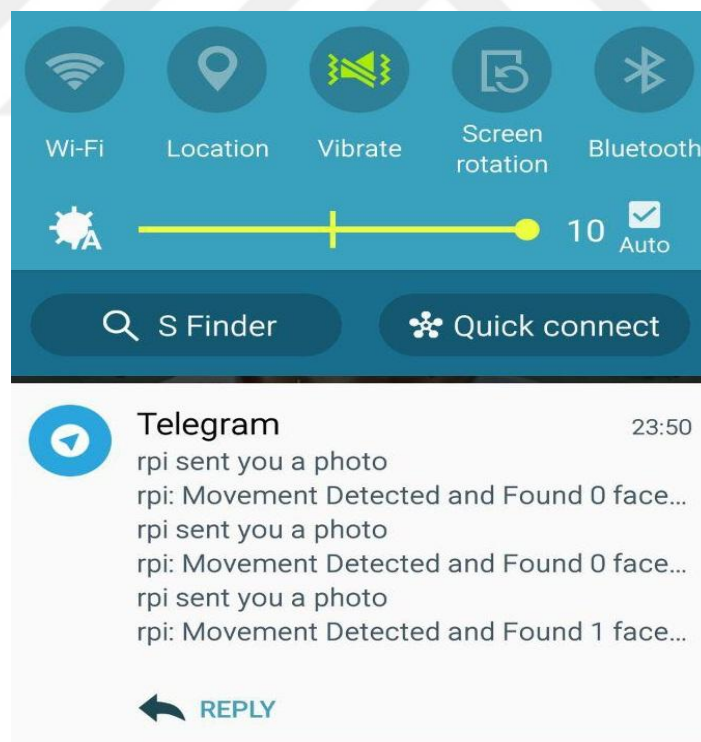
**Figure 3.6.** Experimental setup for face detection application

The Raspberry Pi camera effectively grabs the pictures when any motion is discovered through PIR motion sensor, then face detection is executed. The system was effectively capable to discover the faces in the taken pictures. The algorithm has been used to the entire pictures. The real-time face detection is prepared by means of Cascade Classifier. Viola-Jones face detection algorithm is extra reasonable for real-time face detection. It requires fewer CPU resource and little expenses.

This system applies four operations like find motion, grab pictures, find human faces and sending result and notification to the user's smartphone. The system starts running once running the code. The application will work only when the motion is discovered. Then the output screen displays the message "motion detected" which is shown in Figure 3.7. Simultaneously, the camera grabs the events and the pictures, and notifications will send to a smartphone program. Telegram application is used here to get the pictures and notifications as shown in Figure 3.8. Raspberry Pi 3 has a Wi-Fi wireless technology and Bluetooth. This is helpful to "see activity" and display pictures at once on the smartphone device.

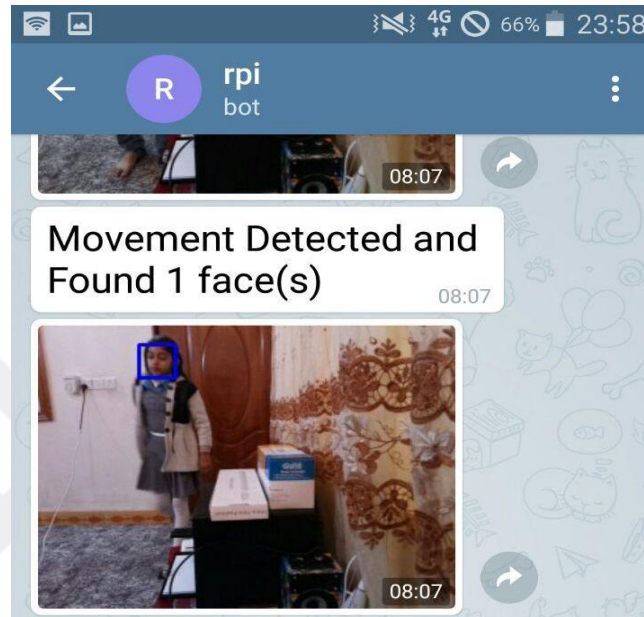


**Figure 3.7.** Show face detection result on the computer screen

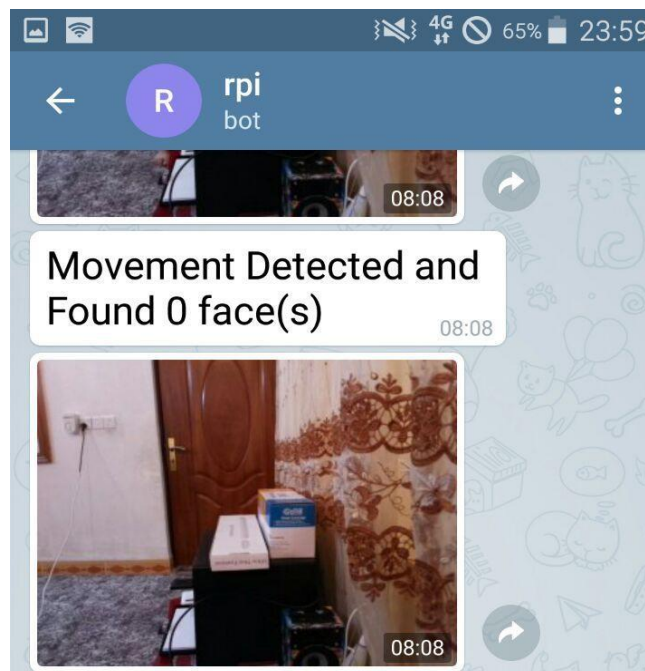


**Figure 3.8.** Screenshot of received notification on the smartphone

The received notifications and results of pictures on the smartphone are shown in Figure 3.9, Figure 3.10, and Figure 3.11.

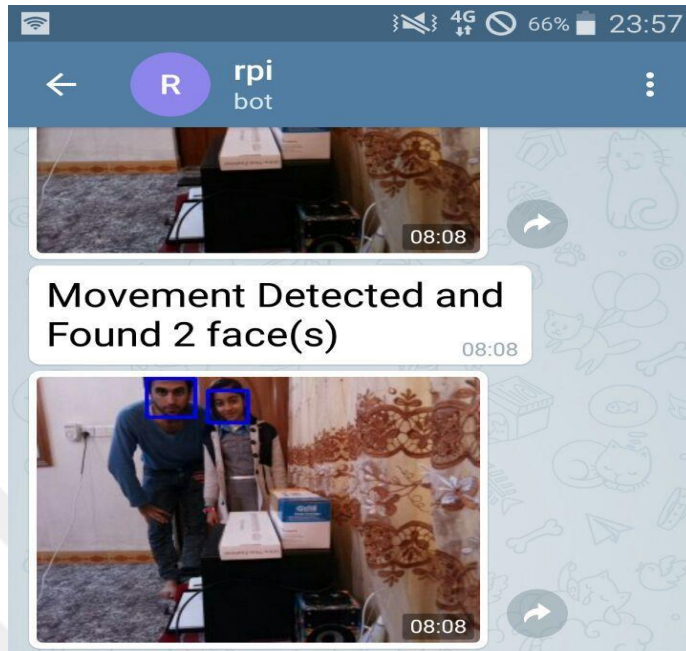


**Figure 3.9.** Result when face is discovered



**Figure 3.10.** Result when face not discovered





**Figure 3.11.** Result when multiple faces discovered

The performance of the proposed method is calculated for Raspberry Pi 3. The calculation results are given in table 3.1.

**Table 3.1.** Time execution and detection rate for face detection

| Number of Images | Time Execution (second) | Detection Rate (%) |
|------------------|-------------------------|--------------------|
| 5                | $4.3 \pm 2.5$           | $100 \pm 0.0$      |
| 9                | $31 \pm 3.7$            | $96.87 \pm 4.3$    |
| 25               | $74 \pm 8.$             | $97.36 \pm 3.7$    |

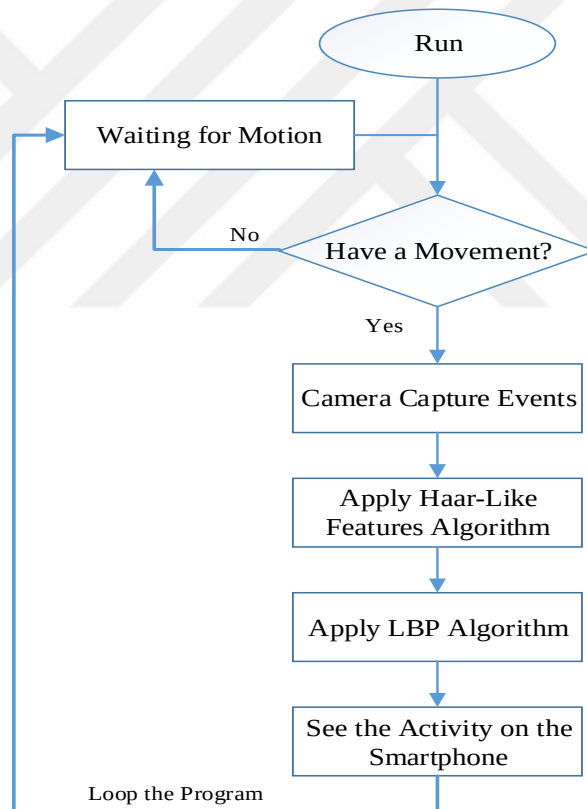
As shown in Table 3.1, the performance of the face detection method is good for Raspberry Pi 3. The time execution consists of motion detection, capturing the image, face detection and sending the image to a smartphone.



In this study, we have outlined embedded face detection with an intelligent security system which is capable to catch picture and mailing them to a smartphone. The real time surveillance of the house is necessary for security application. In this paper, a security system is offered and real time face detection is implemented. The system will alarm the user whether every intruder has gone the office or home. A smartphone is the main gadget of the system that is used by the user to get notifications with the captured pictures. To improve and computerize the security of cities, industries, towns, house and the earth, this system can be utilized. The Pi camera module in particular designed for the Raspberry Pi. For implementing the house automation, the Raspberry Pi 3 is an efficient gadget. It has extremely excellent features and little price embedded equipment platform. Thus, we have used PIR sensor and Pi camera module to catch the motion when detected.

#### 4. AN IOT BASED FACE RECOGNITION FOR FOREIGN PERSON DETECTOR

In the proposed system, a camera is utilized to achieve the image when a movement detected by PIR sensor. Then, computer vision module will apply to the captured images to detect and recognize the human faces. Then, it will send it to a smartphone. This system is very useful and important to secure a place. The algorithm can be separated into three sections, movement detection, face detection and face recognition. If there is no movement detected, the program will not go to face detection and recognition. Figure 4.1, illustrates the flow chart for the proposed system.



**Figure 4.1.** Flowchart for face recognition system

In Figure 4.1, the motion detection module detects any motion by using PIR sensor. Afterward, the algorithm will search for human faces and after detection, face recognition will be processed. Then, the image will be sent to the mobile device.

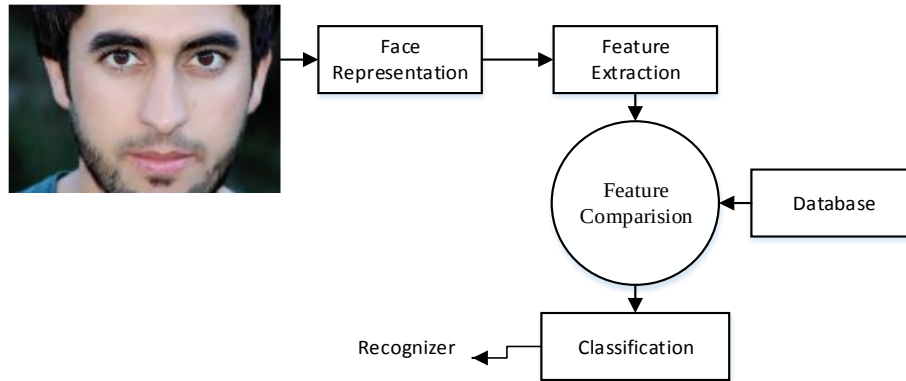
## 4.1 The Proposed Face Recognition Method

Face recognition particularly executes on faces which is one of the pattern recognition tasks. It can be described as classifying a face either known or unknown via comparing a face and putting away known persons in the database. This can be finished by comparing the invariant features got from the strategies that catch the delegate variability of the faces or the structure, the shape and the face attributes such as upper outlines of the eyes, distance between the eye centers, nose and width of eyebrows.

People have an excellent skill to recognize and perceive between faces. But recognizing person's face robotically through a computer is extremely complex. Face recognition procedure utilizes a few other disciplines like computer vision, image processing, neural networks, pattern recognition and psychology.

Face recognition grow to be one of the most active research areas especially in recent years. It has an assortment of large applications in the ranges: public security, access control, credit card verification, criminal identification, law enforcement commerce, information security, human-computer intelligent interaction and digital libraries.

Face recognition system is classified into steps; face detection and face localization according to Haar-like features. Through using weighted LBP algorithm, face features will be extracted [20]. By utilizing face recognition, the procedure of the person identification can be classified into three main stages as illustrated in Figure 4.2. Which they are: face representation, feature extraction and classification [69]. The first task is face representation, that is, how to demonstrate a face. In the face representation task, the face will be represented successive according to algorithms of location and identification. To discover whether the face shows up in the input image or not, face detection must be performed. Locating the position of the face in the captured image will be done in the next step. Via utilizing Haar feature based cascade classifier, face detection and localization is performed [20, 64].



**Figure 4.2.** General steps for identification procedure with face recognition

#### 4.1.1 Face Representation

The initial step, face representation characterizes in how to display a face and check which algorithms which can be utilized for detection and recognition function. In line to obtain computerized face image information via a camera connected to a camera board, Haar-like feature and AdaBoost classifier are embraced to realize face detection [18]. There are a lot of factors of the captured image, like shine, many redundant data, contrast and image size which affect the precision of face recognition system. So, it is more vital to prepare the caught picture before face recognition. For this purpose, the input face converts to grayscale and normalize it.

#### 4.1.2 Feature Extraction

Face image has the most useful and special features which are extracted in the feature extraction stage. Face image with the images from the store will be compared to obtain the features. Feature extraction phase is the core of face recognition system. For the exemplification of the face and to measuring matches among images, these features can be utilized.

There are some techniques in face recognition to achieve and bring the most important features from face images. One of these feature extraction technique called Local Binary Pattern Histogram (LBPH). LBPH technique was produced by Ojala et al. [16]. Describing the shape and texture of a digital image it is conceivable with LBP. This technique provides good

results and efficient for real-time applications. The technique appears to be pretty strong beside face images with various facial expressions, old aging, image rotation and diverse lightening conditions. Haar-like features and LBP are robust when compared to the others. LBP features have performed very well in different applications, including texture classification and segmentation, surface inspection and image retrieval. This LBP technique is finished via partitioning a picture into a number of small regions as appear in Figure 4.3. The features are extracted from every individual region. To portray the surroundings of pixels in the areas, these features are coded into binary patterns. Each area is processed to calculate the features. These calculated features from all areas are combined to a single feature histogram, which forms a demonstration of the picture. Pictures can then be compared through computing the similarity among their histograms.

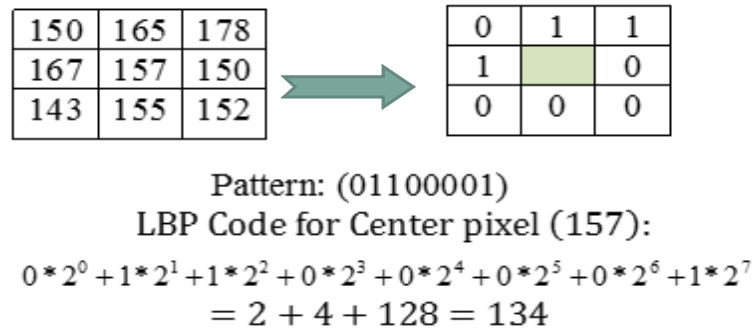


**Figure 4.3.** A preprocessed image divided into 64 areas

In this stage, LBP is calculated for each pixel. Generally, LBP works on grayscale pictures with the 3\*3 neighborhood. First, LBP computes the binary relations among every pixel in the picture and its local neighbor points in the grayscale. After that, the binary relationship is weighted into an LBP code according to a confident set of laws.  $l(xc, yc)$  is any pixel surrounded by a local area of a picture,  $l(xc, yc)$  is the middle of the 3\*3 window, the other 8 points are  $g_0, g_1, \dots, g_7$ . Describing the local area texture as  $T = t(g_c, g_0, \dots, g_7)$ , binary other 8 pixels inside the window utilizing the threshold, here set the threshold for the gray rate of middle point in the window. The LBP code is computed for the middle pixel with coordinate  $(x, y)$  via:

$$T \approx t(s(g_0 - g_c), \dots, s(g_7 - g_c)) \quad (4.1)$$

LBP value computed as demonstrated as follows. There are eight neighboring pixels in the picture. In the grayscale picture, to compute the LBP value for pixels, compare the pixel to all of its eight neighbors on its left-middle, left-top, left-bottom, right-bottom and right-top. When the middle pixel's value is bigger than the neighbor's value, write zero. Something else, write one. This offers an 8-digit binary number. Binary value normally stored in the center pixel location of the output picture like a decimal number. The entire process is illustrated in Figure 4.4. For instance, the current pixel has value 157. The Comparison start from the neighboring pixel where the label zero. The value of the neighboring pixel with label zero is 150. As it is fewer than the current pixel value which is 157, the 0<sup>th</sup>-bit position will reset in the 8-bit binary array to zero. After that, it will be repeated in the counter clockwise direction. The next label positions one have value 165 which is bigger than the current pixel value. So set the first-bit position in the 8-bit binary to one. If the value of the neighboring pixel is equal to the current pixel value, write 1. After that, continue to transfer to the next neighboring pixel pending that reach the eighth neighboring pixel. At that time the 8-bit binary pattern is changed to a decimal number and the decimal number is subsequently put away in the consistent pixel position in the LBP mask.



**Figure 4.4.** Example of LBP calculation

After calculating LBP operator, local features are created through calculating histograms of LBP over local picture areas in local feature extraction process.

When LBP mask computed for every pixel, the feature vector of the picture can be constructed by computing the LBP histogram for each cell. This histogram can be viewed as a 256- dimensional feature vector. After that, the standardization of the LBP histogram and

concatenate histograms of every cell will be completed. This offers a feature vector for the whole picture. The length of the feature vector minimizes from 256 to 59 for a single cell.

As shown in Figure 4.5, a face image is divided into 64 sections. A histogram with every possible label is built for each section. This means that each bin in a histogram characterizes a pattern and holds the quantity of its appearance in the section. The feature vector is subsequently constructed through concatenating the regional histograms to one large histogram. Finally, these histograms are used to measure the match among the pictures via computing the distance among the histograms.

LBP codes of every pixel for an input picture are gathered into a histogram like a texture descriptor. The histogram of the picture LBP (x, y) can be characterized as:

$$LBPH(i) = \sum_{x,y} I\{i, LBP(x, y)\}, \quad i = 0, \dots, n - 1 \quad (4.2)$$

Where n is the number of various labels created via the LBP operator and

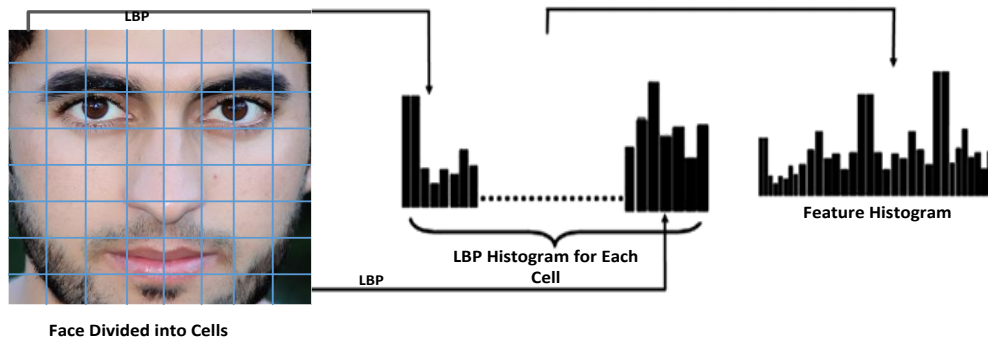
$$I\{A\} = \begin{cases} 1, & A \text{ is true} \\ 0, & A \text{ is false} \end{cases}$$

Later than the LBPH feature vector is extracted, the face recognition is performed via resorting to the K-Nearest Neighbor (KNN) classifier based on the histogram matching methods. For example, the commonly utilized Chi-square measure:

$$d(M, S) = \sum_i (M_i, S_i) \quad (4.3)$$

$$d_{x^2}(M, S) = \sum_{i=1}^B \frac{(M_i - S_i)^2}{M_i + S_i} \quad (4.4)$$

Where M and S is the gallery and explores histogram objects. B is the number of bins in the histogram. To compare two face pictures, a sample (S) and a model (M), the diversity among the feature vectors have to be measured. This can be done with histogram features.

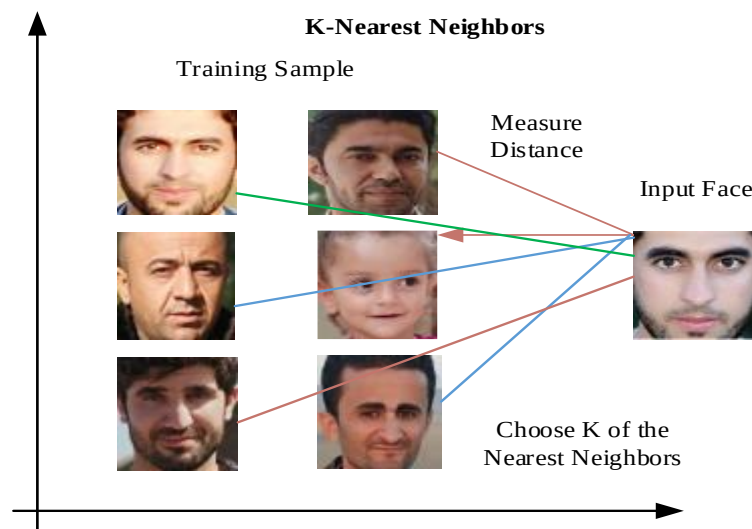


**Figure 4.5.** An example of extracting LBPH feature vector

### 4.1.3 Classification

Face image with the images from the store will be compared to obtain the features in the classification stage. The comparison is performed utilizing the local features acquired in the earlier phase of the algorithm. According to the face image identity from the store, the maximum matching score is going to be the output of the classification portion, in this case, with the small variations compared to the input face image. Likewise, the value of the threshold can be utilized to determine if the variations are small sufficiently.

After the distance between features calculated by Chi-square function, then KNN classifier will apply to identify faces. KNN is a data classification technique that can be utilized in the face recognition system. It is a strong classification method utilized in pattern recognition. KNN store every accessible faces and classifies new faces based on a similarity measure. In KNN, input face will be compared with each example of training data. The meaning of the nearest neighbor technique is the select of uniqueness faces. It works based on the idea that is minimum distance from the query case to the training examples in order to determine the KNN. Figure 4.6, illustrates the schematic diagram of nearest neighbor.



**Figure 4.6.** Graphic diagram of KNN classification



All pixel in face, denotes unique data. This study recognized face based on all pixel classification. In recognition, before classification stage, pixel matrix of face image should be reforming into a vector. This algorithm, an object is classified by a majority vote of its neighbors. The input face is consequently assigned to the training set that is most common between its KNN, where  $K$  is a positive integer that is normally small. At the end, it is the search for a face that has not existed in the database at all [70-71].

## 4.2 Experimental Results

Smartphone application was integrated with the proposed system to develop a smart movement and face detection and recognition for personal area. The experimental hardware setup of the proposed system is given in Figure 4.7. Apart from the hardware configuration, the required software should also be installed. In the proposed system, python script is used and installed Raspbian OS which is good with the Raspberry Pi.

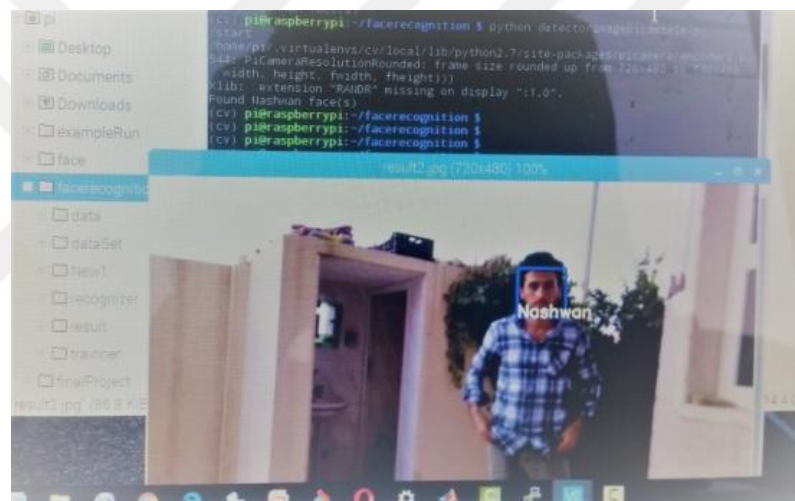


**Figure 4.7.** Experimental setup for face recognition

The Raspberry Pi camera successfully captures the image when PIR sensor detects any movement. After that, face detection and recognition are implemented. The system was able to successfully identify the faces in the captured images. The algorithm has been applied to all the images. The real-time face detection is done by means of Haar-like features and real-time

face recognition is done by means of local binary pattern (LBP). The face detection and recognition algorithm are more reasonable for real-time because they need less CPU resource and low costs.

This application performs four operations such as motion detection, capturing images, face detection and recognition with sending output images and notifications to the user's smartphone. The setup begins working after importing and running the code. The system will be activated just when the movement is detected. When the movement is detected, the output screen shows the message “motion detected” and recognizes face which is appeared in Figure 4.8.



**Figure 4.8.** Display output on the PC screen

In Figure 4.8, the implementation of the proposed face detection and recognition system is shown on a PC screen. For this purpose, the Raspberry Pi is connected to a monitor. At the same time, the camera captures the events. The notifications and the images will be sent to a smartphone application “Telegram” as shown in Figure 4.9. Raspberry Pi 3 has Wi-Fi wireless technologies and Bluetooth. This is useful to view activity and show images immediately on the smartphone gadgets.



**Figure 4.9.** Screenshot of received notification on the smartphone for face recognition

The received output images and notification on the smartphone are given in Figure 4.10.



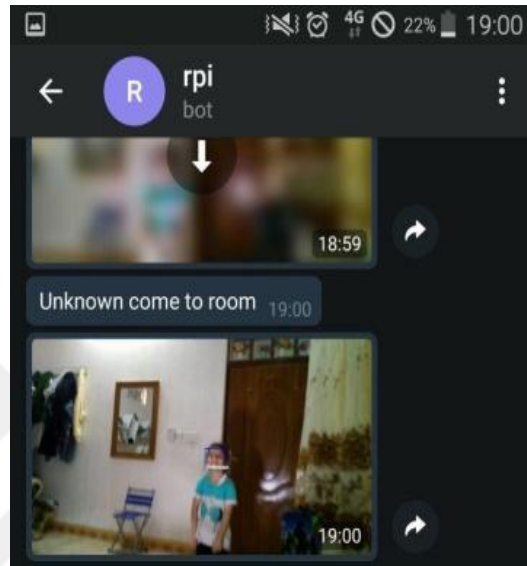
(a) wafa face



(b) nashwan face

**Figure 4.10.** The output of the algorithm for different face detection and recognition

In Figure 4.10, the algorithm detects and recognizes human faces successfully for known faces that stored in the database. If unknown face enters the room, the system will notify unknown person is coming to the room as appear in Figure 4.11.



**Figure 4.11.** The output of the algorithm for unknown face

Currently, OpenCV gives three techniques for face recognition: Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Local Binary Patterns Histograms (LBPH).

In this thesis, LBPH is used for face recognition. Also, the result of LBPH algorithm is compared with PCA and LDA algorithm. PCA is well called Eigenfaces and LDA called Fisherfaces. The training set is used to build the recognition once the algorithm is requested to recognize some unidentified face. All of the three previously mentioned techniques utilize the training set a bit differently.

Both PCA and LDA techniques are one of the well-known techniques for face recognition. They are known to be very sensitive to pixel level variations like facial expression, illumination and pose variations. PCA and LDA find a mathematical description of the most predominant features of the training set such as a whole. PCA and LDA are both affected by brightness in actual life.

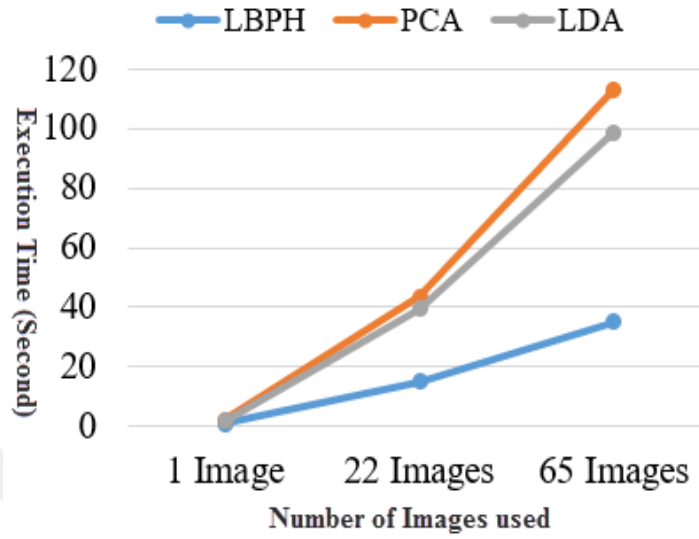
PCA procedure considers illumination such as a vital feature. PCA looks at the training faces of each person at once and discovers main components from each of them. Via doing that, it doesn't concentrate on the features that separate one individual from another. Instead, it focuses on the ones that represent every face of each person in the training data such as a whole. Exactly, LDA face recognizer method extracts principal components that separate one person from the others. In that sense, a person's components do not dominate the others.

LBPH is a relatively new technique, mainly advanced for texture analysis. It analyzes every face in the training set independently and separately. While the PCA technique looks at the dataset as a whole, the LBPH technique is fairly less complex. In the LBPH technique, every picture in the dataset is characterized locally. LBPH will perhaps implement better in various situations and light conditions. Nevertheless, it will depend on training and testing data sets. Compared to PCA and LDA techniques, LBPH method gives better result under lighting and pose variations. LBPH accomplishes the highest accuracy in various probe sets in the bellow tests.

In the proposed system, the Raspberry Pi 3 platform is used. Raspberry Pi 3 is based running at 1.2GHz, 64-bit quad-core processor with 1 GB RAM. Table 4.1 and Figure 4.12, shows time execution comparison for each of three algorithms on the Raspberry Pi 3.

**Table 4.1.** Time execution comparison for face recognition algorithms

| Method | 1 image (seconds) | 22 images (seconds) | 65 images (seconds) |
|--------|-------------------|---------------------|---------------------|
| LBPH   | $0.71 \pm 2.8$    | $15.04 \pm 4.3$     | $35.21 \pm 9.6$     |
| PCA    | $2.07 \pm 4.5$    | $43.58 \pm 6.9$     | $113.17 \pm 11.2$   |
| LDA    | $1.93 \pm 4.2$    | $39.44 \pm 5.8$     | $98.90 \pm 10.6$    |



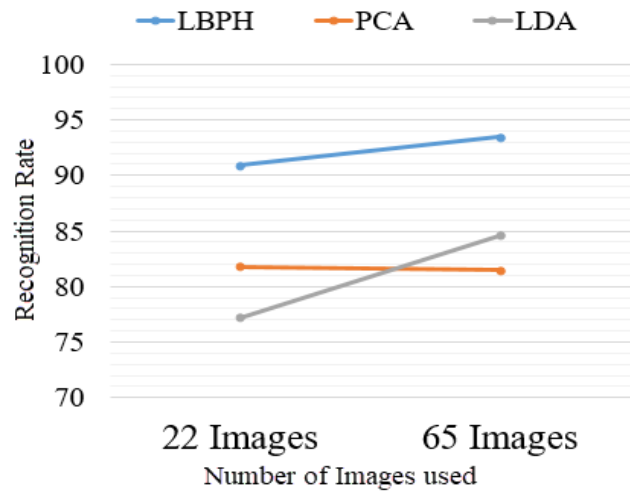
**Figure 4.12.** Time execution comparison graph for face recognition algorithms

Table 4.1 and Figure 4.12 compare the time execution of three face recognition algorithms. As shown, the proposed LBPH algorithm outperforms from another. Generally, LBPH algorithm robustness and makes the best performance result.

The accuracy of each algorithm is shown in Table 4.2 and Figure 4.13.

**Table 4.2.** Detection rate comparison for face recognition algorithms

| Method | Test(1) 22 image |                 |                  | Test (2) 65 image |                 |                  |
|--------|------------------|-----------------|------------------|-------------------|-----------------|------------------|
|        | True Recognize   | False Recognize | Detection Rate % | True Recognize    | False Recognize | Detection Rate % |
| LBPH   | 20               | 2               | 90.90±0.9        | 61                | 4               | 93.8±1.7         |
| PCA    | 18               | 4               | 81.80±2.9        | 53                | 12              | 81.5±3.1         |
| LDA    | 17               | 5               | 77.02±2.3        | 55                | 10              | 84.6±2.6         |

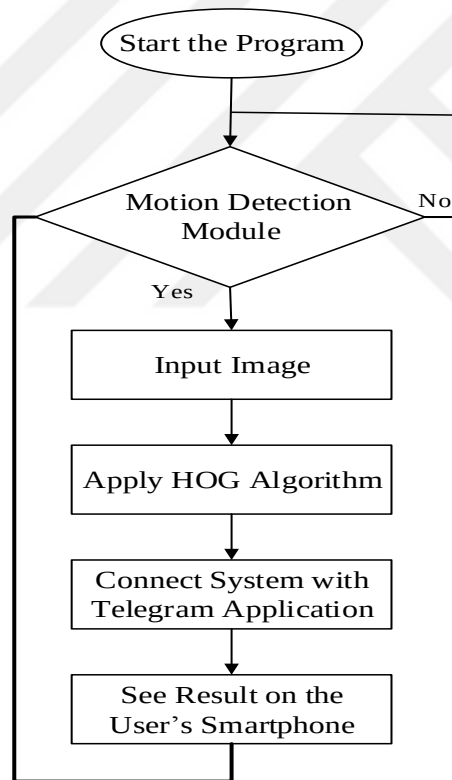


**Figure 4.13.** Accuracy comparison graph for face recognition algorithms

As seen in Table 4.2 and Figure 4.13, the accuracy of the LBPH algorithm is better than PCA and LDA. Also, PCA is better than LDA, but when the number of a sample is large, LDA outperforms PCA.

## 5. HOG BASED BODY DETECTION AND ITS IOT IMPLEMENTATION ON RASPBERRY PI 3

In the offered system, using a camera to achieve the image when a movement is detected through PIR sensor. Then, computer vision module will apply to the captured images to detect the humans. Finally, it will send it to a smartphone. This system is very useful and important to secure a place. The algorithm can be separated into two sections which are: movement detection and human detection. If there is no movement detected, the program will not go to human detection. Figure 5.1, illustrates the flow chart for the proposed system.



**Figure 5.1.** Flowchart for body detection system

In Figure 5.1, the motion detection module detects any motion by using PIR sensor. Afterward, the human is searched in the detected image. If the human is detected in the image, the image will be sent to the mobile device.



## 5.1 The Proposed Method for Body Detection

Human detection is utilized in different applications. Body features are detected and others are ignored. Histogram of Oriented Gradients (HOG) is the most popular algorithm for human detection. It is utilized to detect single or multiple humans. OpenCV library is utilized for human detection because they are the best open source library obtainable to perform image processing and they are not too much complex. Dalal and Triggs [51] proposed an efficient method for the human detector. They utilize a dense grid of HOG and find the humans in terms of a Support Vector Machine.

In practice, an image is densely examined by a detection window which moves to any conceivable position and with any conceivable scale, searching for humans. The task is then to build a superior classifier that can separate humans and non-humans in a short time and with low false positive since an image can contain from  $10^4$  to  $10^5$  detection windows relying on the intensity of the search [72].

There are two main steps for human detection: features extraction and classification learning. The first algorithm consists of extracting the most important information from the data accessible by applying HOG algorithm and the second algorithm to classify humans by applying SVM algorithm.

HOG is a kind of feature descriptor algorithm. The whole body is required to be detected. HOG algorithm provides a good representation of borders and silhouettes. Also, it is invariant to small image movement and light difference changes. It can be processed in a constant time independently of their size.

HOG feature descriptor is computed for a  $64 \times 128$  detection window of an image and return a vector of size 3780 for human detection. The image could be encoded in five stages: pre-processing and normalizing of the image, computing gradients, weighted vote into spatial and orientation cells, contrast normalize across overlapping spatial blocks, and HOG across detection window. The image could be preprocessed to get the gradient and orientation images which are required to compute the HOG in the first two stages. The last three steps are very important parts of HOG [73, 74]. The stages for computing the HOG descriptor for a  $64 \times 128$  patch is given as follows.

The first step is calculating HOG features which are necessary to normalize and preprocess the image for a fast HOG calculation. For human detection, an image might be of any size but HOG feature descriptor is computed on a  $64 \times 128$  patch of an image. For example, to illustrate this step, an image of size  $625 \times 451$  will be processed. A patch of size  $193 \times 399$  has been selected for computing HOG feature descriptor. This patch will crop out of an image and resize to  $64 \times 128$ .

Also, the similar result could be achieved after gamma normalization. Gamma normalization is an image enhancement technique. The contrast of images could be improved by gamma normalization. It is well done by:

$$y(n_1, n_2) = x^\gamma(n_1, n_2) \quad (5.1)$$

In (1),  $x(n_1, n_2)$  equal to the input image, and  $y(n_1, n_2)$  equal to the output image. Good detection performance will be produced when the parameter  $\gamma$  set to 0.5. The preprocessing and gamma normalization are shown in Figure 5.2.



**Figure 5.2.** Preprocessing image and gamma normalization

In Figure 5.2, a part of the image is cropped and gamma normalization is applied to the cropped image. The normalized image gives a superior contrast that is very helpful for the next processing. The second stage is to calculate the gradient scales and orientations using the horizontal and vertical convolution masks  $[-1, 0, 1]$  and  $[-1, 0, 1]^T$  in the sub-image covered

by detection window. HOG descriptor is based on the oriented gradient. The bellow equations are utilized to calculate the gradient:

$$y_i(n_1, n_2) = h_i(n_1, n_2) * x(n_1, n_2), i = 1, 2 \quad (5.2)$$

In (2),  $h_i(n_1, n_2)$  is the derivative mask and it taken as:

$$h_1(n_1, n_2) = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}, h_2(n_1, n_2) = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \quad (5.3)$$

$h_1(n_1, n_2)$  and  $h_2(n_1, n_2)$  is utilized to calculate the horizontal gradient and vertical gradient respectively. Also, the overall gradient and orientation image can be obtained as:

$$y_m(n_1, n_1) = \sqrt{y_1^2 + y_2^2}; y_\theta(n_1, n_1) = \tan^{-1}(y_2/y_1) \quad (5.4)$$

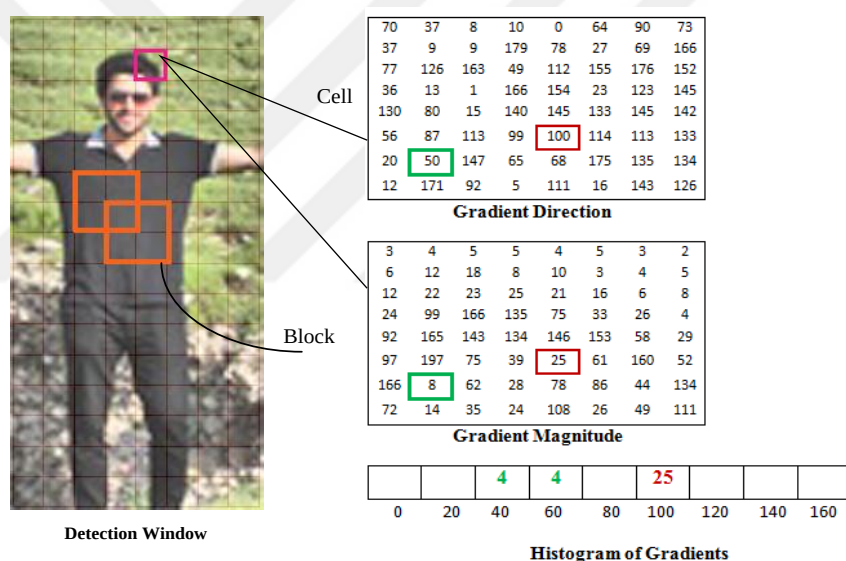
A lot of inconsiderable data is removed by the gradient image such as highlighted outlines and constant shaded background. The gradient image is shown in Figure 5.3 for a human.



**Figure 5.3.** Computing the gradients

The image is divided into  $8 \times 8$  cells and a histogram of gradients is computed for every cell in the third step. Computing the weighted vote of each pixel to the orientation histogram which has several bins evenly spaced over  $0^\circ \sim 180^\circ$  in each cell divided from the sub-image and then accumulate all votes as the cell's feature vector. At every pixel, an  $8 \times 8$  cell knowing the gradient direction and gradient magnitude, in this case, there are 64 directions and 64 magnitudes which are 128 numbers. Next, these 128 numbers will change to a 9-bin histogram. Gradients directions 0, 20, 40, 60, 80, 100, 120, 140 and 160 degrees corresponded by the binaries of the histogram. Binaries are chosen depends on the direction, and the value

that goes into the binaries which is called vote, is chosen depends on the magnitude. For one binaries or two binaries in the histogram, every pixel votes. A vote equivalent to the magnitude of the gradient is cast through the pixel to the bin if the direction of the gradient at a pixel is properly 0, 20 ...160 degrees. A pixel divides its vote between the two nearer bins depends on the range of the binaries if the direction of the gradient is not properly 0, 20 ...160 degrees. For example, if a pixel with the magnitude of the gradient is 25 and the angle is 100 degrees will vote for the third binaries with price 25. So, if a pixel with gradient 8 and angle 50 will vote 4 for both the third binaries (angle 40) and the fourth binaries (angle 60). Figure 5.4 illustrates the computation of HOG.



**Figure 5.4.** Computing histogram of gradient oreintation

In Figure 5.4, HOG calculation is shown for one block. The commitments of each pixel in the 8\*8 cells are added upward to make the 9-binaries histogram. Sequentially, after computing histogram across an 8\*8 cell, the image will be normalized across a 16\*16 block. Collect the cells into larger overlapping blocks and each of it contains several cells. After that, Normalizing a vector of length 3. For example, an RGB color vector [128, 64, and 32] will be computed as  $\sqrt{128^2 + 64^2 + 32^2} = 146.64$ .

Dividing all components of this vector via 146.64 gives us a normalized vector [0.87, 0.43, and 0.22]. Normalizing a vector will remove the scaling. Each 16\*16 block has four

histograms which can be linked to form a  $36 * 1$  component vector and it can normalize like a  $3*1$  vector. The window is then moved via 8 pixels and a normalized  $36*1$  vector computes across this window and the procedure is repeats all over again. Finally, after computation and normalization of the histogram, computation of the final component vector for the image will be processed. For this purpose, achieve the descriptor containing each component of the normalized cell reactions from each of the blocks in the detection window. There are seven horizontal and fifteen vertical positions making an aggregate of  $7*15$  blocks for the  $64*128$  detection window, and each  $16 \times 16$  block is represented through a  $36*1$  vector.

After the feature extraction of each training samples, the detector is trained by linear Support Vector Machine (SVM). SVM is one of the generality supervised binary classification algorithms. The image will scan at various scales. Then, HOG feature is computed in each sub-image. Also, it uses the learned SVM detector to build a classified image: human or non-human.

## 5.2 Experimental Results

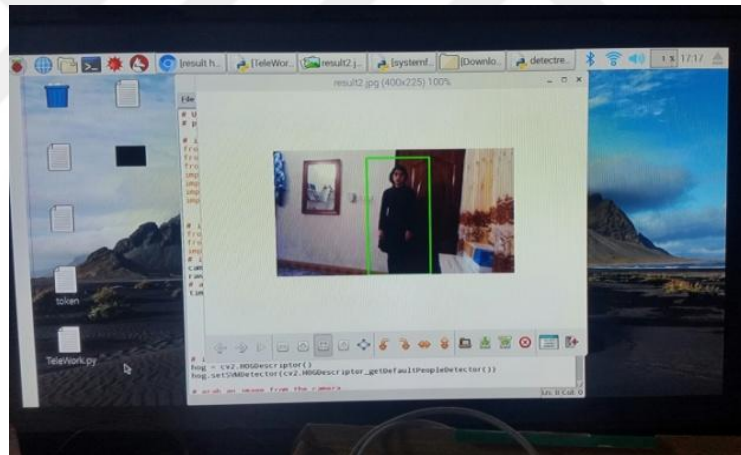
Smartphone application was integrated with the proposed system to develop a smart movement and human detection for offices and houses. The experimental hardware setup of the proposed system is given in Figure 5.5. Apart from the hardware configuration, the required software should also be installed. Python script utilized in the proposed system and Raspbian OS installed on the Raspberry Pi. Since it's the best operating system for Raspberry Pi.



**Figure 5.5.** Experimental setup for body detection

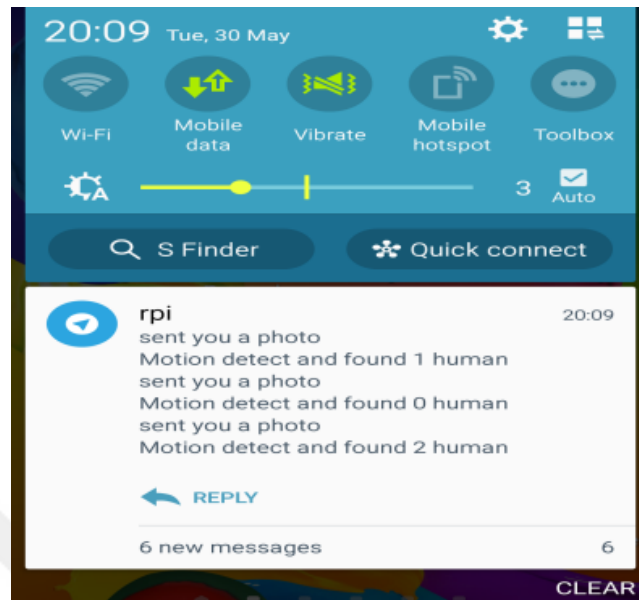
The Raspberry Pi camera has successfully captured the image when any movement is detected by PIR sensor. After that, human detection is implemented. The system was able to successfully identify the humans in the captured images. The algorithm has been applied to all the images. The real-time human detection is done by means of HOG. The human detection algorithm is more reasonable for real-time human detection because it needs less CPU resource and low costs.

This application performs four operations such as motion detection, capturing images, human detection and sending output images and notifications to the user's smartphone. The setup begins working after importing and running the code. The system will be activated just when the movement is detected. When the movement is detected, the output screen shows the message “motion detected” which is appeared in Figure 5.6.



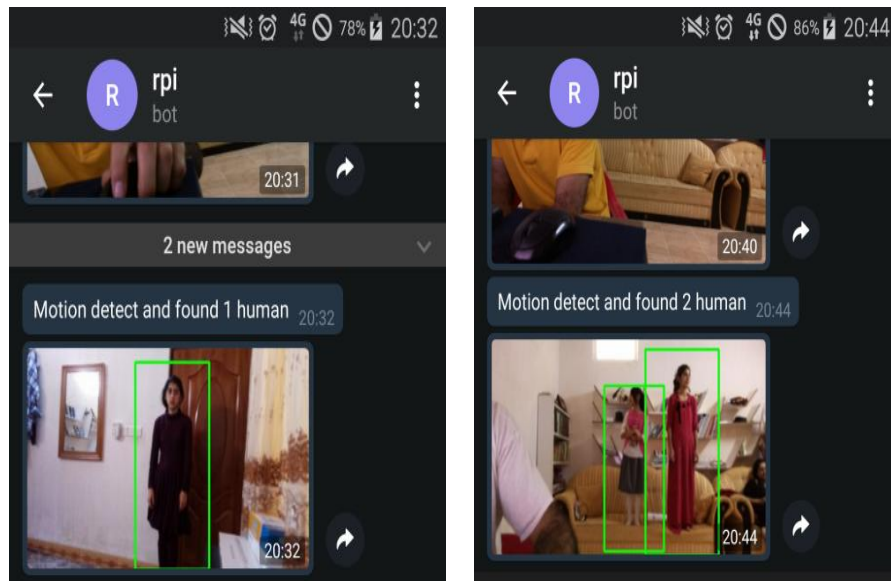
**Figure 5.6.** Display body detection output on the PC screen

In Figure 5.6, the implementation of the proposed human detection system is shown on a PC screen. For this purpose, the Raspberry Pi is connected to a monitor. At the same time, the camera captures the events. Later, the notifications and the images will be sent to a smartphone application “Telegram” as shown in Figure 5.7. Raspberry Pi has Wi-Fi wireless technologies and Bluetooth. This is useful to view activity and show the images immediately on the smartphone gadgets.



**Figure 5.7.** Screenshot of received notification on the smartphone for body detection

The received output images and notifications on the smartphone are given in Figure 5.8.



(a) One human detection

(b) Multiple human detection

**Figure 5.8.** The output of the algorithm for one and multiple human detections

In Figure 5.8, the algorithm successfully detects both single and multiple humans. The time execution and the detection rate performance of the proposed method are given in Table 5.1.

**Table 5.1.** The detection rate and time execution for body detection on Raspberry Pi 3

| Detection Rate (%) | Time Execution (second) |
|--------------------|-------------------------|
| 93.89±5.3          | 0.41±0.09               |

As shown in Table 5.1, the detection rate and time execution of the proposed method is good for real-time applications. The results show that the proposed scheme can be applied in an embedded platform.

The real-time monitoring of the home is required for a security system. In this study, a security system is proposed and a real-time human detection is implemented. It will alert the client if any person has entered the house or office. The smartphone is the main device of the system that is utilized by the client to obtain notifications with the captured images. This system can be used to enhance and automate the security of our industries, cities, homes and towns. Raspberry Pi 3 has very good features and low cost embedded hardware platform. Therefore, we have utilized PIR sensor and Raspberry Pi camera module to capture the activity when any movement is detected.



## 6. CONCLUSIONS

In this thesis, an embedded face detection and face recognition with smart security system are designed to be able to capture an image and send it to a smartphone. So, when a face is detected and recognized, the system will notify the user by using a smartphone and displays who is he in that area. By adding the face recognition system, people will be easily recognized and a safer city will be built. Also, offering a possible solution to utilize computer vision in the IoT in this thesis. This could permit utilizing Raspberry Pi cameras and sensors. It will alert the client if an intruder has entered the house or the office. Smartphone is the main benefit of this thesis which is utilized by the client to obtain notifications with the captured images. For executing the home automation, Raspberry Pi proves to be an effective, smart and economic platform. So, showing that the PIR sensor and Raspberry Pi camera module can be utilized to capture the activity. Raspberry Pi 3 has very good features and low cost embedded hardware platform.

Computer vision could simplify the use of the IoT in everyday life and make new paths to communicate with our environment. This system is helping to enhance and automate the security of industries, cities, homes, towns and earth. Low accuracy is one of the properties of computer vision. In this thesis, OpenCV is used because it is the superior library for image processing applications.

In this thesis, LBPH algorithm is used to recognize faces. Also, the result of LBPH algorithm is compared with PCA and LDA algorithm. The results show that LBPH method gives better results under lighting and pose variations. LBPH accomplishes the highest accuracy in various probe sets in tests.

On the other hand, the real-time body detection is important for the security system. For this purpose, in this thesis, a real-time human detection is implemented. It will alert the client if any person has entered the house or office.

## REFERENCES

- [1] **C. González García, D. Meana-Llorián, B.C.P. G-Bustelo, J.M.C. Lovelle**, A review about smart objects, sensors, and actuators, *Int. J. Interact. Multimed. Artif. Intell.* 4 (2017) 7–10. <http://dx.doi.org/10.9781/ijimai.2017.431>.
- [2] **A.J. Jara, Y. Sun, H. Song, R. Bie, D. Genoud, Y. Bocchi**, Internet of things for cultural heritage of smart cities and smart regions, in: 2015 IEEE 29th Int. Conf. Adv. Inf. Netw. Appl. Work, IEEE, Gwangju, 2015, pp. 668–675. <http://dx.doi.org/10.1109/WAINA.2015.169>.
- [3] **H. Gu, D. Wang**, A content-aware fridge based on RFID in smart home for homehealthcare, in: *Adv. Commun. Technol. 2009. ICACT 2009. 11th Int. Conf.*, Phoenix Park, 2009, pp. 987–990.
- [4] **S. Luo, H. Xia, Y. Gao, J.S. Jin, R. Athauda**, Smart fridges with multimedia capability for better nutrition and health, in: 2008 Int. Symp. Ubiquitous Multimed. Comput, IEEE, Hobart, ACT, 2008, pp. 39–44. <http://dx.doi.org/10.1109/UMC.2008.17>.
- [5] **C. González García, D. Meana-Llorián, Pelayo G-Bustelo, Cueva Lovelle, Garcia-Fernandez** Midgar: Detection of people through computer vision in the Internet of Things scenarios to improve the security in Smart Cities, Smart Towns, and Smart Homes. *Future generation Computer Systems* (2017). <http://dx.doi.org/10.1016/j.future.2016.12.033/>.
- [6] **M. Rothensee**, A high-fidelity simulation of the smart fridge enabling productbased services, in: 3rd IET Int. Conf. Intell. Environ, (IE 07), IEE, 2007, pp. 529–532. <http://dx.doi.org/10.1049/cp:20070420>.
- [7] **E. Borgia**, The Internet of things vision: Key features, applications and open issues, *Comput. Commun.* 54 (2014) 1–31. <http://dx.doi.org/10.1016/j.comcom.2014.09.008>.
- [8] **C. Hao, X. Lei, Z. Yan**, The application and implementation research of smart city in China, in: *Syst. Sci. Eng. ICSSE, 2012 Int. Conf.*, Dalian, Liaoning, 2012, pp. 288–292. <http://dx.doi.org/10.1109/ICSSE.2012.6257192>.
- [9] **P. N. Belhumeur, J.P. Hespanha, and D. Kriegman**, “Eigenfaces vs. fisherfaces: recognition using class specific linear projection,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 7, pp. 711–720, 2007.
- [10] **C Gonzalo Sánchez-Arias, Cristian González García, B. Cristina Pelayo G-Bustelo** Midgar: Study of communications security among Smart Objects using a platform of heterogeneous devices for the Internet of Things. *Future generation Computer Systems* (2017). <http://dx.doi.org/10.1016/j.future.2017.01.033>.

- [11] **Ingemar J. Cox, J. Ghosn, and P.N. Yianilos**, “Feature- based face recognition using mixture-distance,” in Computer Vision and Pattern Recognition, 2006. Proceedings CVPR, 2006 IEEE Computer Society Conference on, pp. 209–216.
- [12] **T. Ojala and M. Pietikainen**. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, IEEE Trans on Pattern Analysis and Machine Intelligence, Vol. 24. No.7, July, 2002.
- [13] **Ahonen, T., Hadid, A., Pietikäinen, M.:** Face description with LBP: Application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell. 28(12), 2037–2041 (2006).
- [14] **Jian Yang, D. Zhang, Jing-Yu Yang and A.F. Frangi**, “Two-dimensional pca: a new approach to appearance- based face representation and recognition,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 26, no. 1, pp. 131–137, 2004.
- [15] **Arundhati Das, Mameeta Pukhrambam, Ashim Saha**, "Real-time robust face detection and tracking using extended haar functions and improved boosting algorithm", 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), vol. 00, no., pp. 981-985, 2015, doi:10.1109/ICGCIoT.2015.7380605.
- [16] **T. Ojala, M. Pietikainen and D. Harwood**, “A comparative study of texture measures with classification based on feature distributions” Pattern Recognition vol. 29, 1996.
- [17] **T. Ahonen, A. Hadid and M. Pietikainen**, “Face description with Local Binary Patterns”, Application to Face Recognition. University of Oulu, Finland, 2006, Machine Vision Group.
- [18] **T. Ahonen, A. Hadid, M. Pietikainen and T. Mäenpää**. “Face recognition based on the appearance of local regions”, In Proceedings of the 17th International Conference on Pattern Recognition, 2004.
- [19] **R. Gottumukkal and V.K. Asari**, “An Improved Face Recognition Technique Based on Modular PCA Approach” Pattern Recognition Letters, vol. 25, pp. 429- 436, Mar. 2004.
- [20] **R. Lienhart and J. Maydt**, “An extended set of haar-like features for rapid object detection,” in Image Processing. 2002. Proceedings. 2002 International Conference on, 2002, vol. 1, pp. I–900–I–903 vol.1.
- [21] **Fabrice Bourel & C.C. Chibelushi’s** “Facial Expression Recognition:A Brief tutorial Overview” , Staffordshire University, School of Computing, Beaconside, Stafford ST18 0DG, 101 quartier Boieldieu, La Défense 8, F92042 Paris La Défense Cedex, France.

- [22] **Taranpreet Singh Ruprah's** "Face Recognition Based on PCAAlgorithm", Special Issue of Informatics (IJCSI), ISSN (PRINT) & International Journal of Computer Science: 2231–5292, Vol.- II, Issue-1, 2.
- [23] **Kumar Rahul's & Suman Kumar Bhattacharyya** "Face Recognition by Linear Discriminant Analysis", International Journal of Communication Network Security, ISSN: 2231 – 1882, Volume-2, Issue-2, 2013.
- [24] **S'ébastien Marcel, Guillaume Heusch's and Yann Rodriguez** "On the Recent Use of Local Binary Patterns for Face Authentication", International Journal of special issue on Facial Image Processing, Image and Video Processing, October 20, 2006.
- [25] **Md. Abdur Rahim, Tanzillah Wahid, Md. Najmul Hossain & Md. Shafiul Azam:** Face Recognition using Local Binary Patterns, Global Journal of Computer Science and Technology Graphics & Vision, Volume 13 Issue 4 Version 1.0 Year 2013.
- [26] **Zhao, W., Chellappa, R., Rosenfeld, A., Phillips, P.J.:** Face recognition: a literature survey. Technical Report CAR-TR-948, Center for Automation Research, University of Maryland (2002).
- [27] **Y. Pang, Y. Yuan, X. Li, J. Pan,** Efficient HOG human detection, Signal Process. 91 (2011) 773–781.
- [28] **W. Lorenzo, R. Gonzalez-Crespo, A. Castillo-Sanz,** A prototype for linear features generalization, Int. J. Interact. Multimed. Artif. Intell. 1 (2010) 59. <http://dx.doi.org/10.9781/ijimai.2010.1310>.
- [29] **Ms. RenukaChumurkar, Prof. Vijay Bagdi,** "Smart Surveillance Security &Monitoring System Using Raspberry PI and PIR Sensor", International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-2, Issue-1, January 2016 ISSN: 2395-3470.
- [30] **Md. Nasimuzzaman Chowdhury, Md. ShibleeNooman, SrijonSarke,** "Access Control of Door and Home Security by Raspberry Pi Through Internet", International Journal of Scientific & Engineering Research, Volume 4, Issue 11, November-2013 ISSN 2229-5518.
- [31] **Sushma.N.Nichal, Prof.J.K.Singh,** "Raspberry pi Based Smart Supervisor using Internet of Things (IoT)",International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE) Volume 4, Issue 7, July 2015, ISSN: 2278 – 909X.

- [32] **Nashwan A.Othman, Ilhan Aydin**, A New IoT Combined Face Detection of People by Using Computer Vision for Security Application. IEEE International Conference on Artificial Intelligence and Data Processing (IDAP17), September 16-17, Malatya, Turkey, 2017.
- [33] **Nashwan A.Othman, Ilhan Aydin**, A New IoT Combined Body Detection of People by Using Computer Vision for Security Application. IEEE International Conference on Computational Intelligence and Communication Networks (CICN 2017), September 16-17, Girne, Cyprus, 2017.
- [34] **J.A. Stankovic**, Research directions for the Internet of things, IEEE Internet Things J. 1 (2014) 3–9. <http://dx.doi.org/10.1109/JIOT.2014.2312291>.
- [35] **T. Polk, S. Turner**, Security challenges for the Internet of things, in: Work. Interconnecting Smart Objects with Internet, Prague, 2011. <https://www.iab.org/wp-content/IAB-uploads/2011/03/Turner.pdf>.
- [36] **K.T. Nguyen, M. Laurent, N. Oualha**, Survey on secure communication protocols for the Internet of things, Ad Hoc Networks (2015) <http://dx.doi.org/10.1016/j.adhoc.2015.01.006>.
- [37] International Telecommunication Union, Overview of the Internet of things, 2012, 14. <https://www.itu.int/rec/T-REC-Y.2060-201206-I>.
- [38] Vienna University of Technology, European Smart Cities, 2015.<http://www.smart-cities.eu> (accessed 28.09.16).
- [39] **P. Martinez-Julia, E.T. Garcia, J.O. Murillo, A.F. Skarmeta**, Evaluating video streaming in network architectures for the Internet of things, in: 2013 Seventh Int. Conf. Innov. Mob. Internet Serv. Ubiquitous Comput., IEEE, Taiwan, 2013, pp. 411–415. <http://dx.doi.org/10.1109/IMIS.2013.76>.
- [40] **Shiu Kumar**, “Ubiquitous Smart Home System using Android Application ”in IJCNC, vol 6,Jan 2014.
- [41] **R. Bayindir, , I. Colak, , G. Fulli, , K. Demirtas**, Smart grid technologies and applications. Renewable and Sustainable Energy Reviews, Volume 66, December 2016, Pages 499-516. <http://dx.doi.org/10.1016/j.rser.2016.08.002>.

- [42] **C. González García, J.P. Espada, E.R.N. Valdez, V. García-Díaz**, Midgar: Domain specific language to generate smart objects for an Internet of things platform, in: 2014 Eighth Int. Conf. Innov. Mob. Internet Serv. Ubiquitous Comput, IEEE, Birmingham, United Kingdom, 2014, pp. 352–357. <http://dx.doi.org/10.1109/IMIS.2014.48>.
- [43] Carriots, 2011. <https://www.carriots.com/> (accessed 29.07.15).
- [44] ThingWorxTM, ThingWorx, 2015. <http://www.thingworx.com/> (accessed 29.07.15).
- [45] Nimbits Inc., Nimbits, 2015. <http://www.nimbits.com> (accessed 29.07.15).
- [46] IoBridge, Thingspeak, 2013. <http://www.thingspeak.com> (accessed 29.07.15).
- [47] CyberVision Inc., KAA, 2014. <http://www.kaaproject.org> (accessed 29.07.15).
- [48] TelegramBots Inc., TelegramBotAPI, 2016. <https://core.telegram.org/bots> (accessed 21.11.16).
- [49] **Ming Tao, Jinglong Zuo, Zhusong Liu, Aniello Castiglione, Francesco Palmieri**, Multi-layer cloud architectural model and ontology-based security service framework for IoT-based smart homes. Future Generation Computer Systems, 21 November 2016. <http://dx.doi.org/10.1016/j.future.2016.11.011>.
- [50] **C. Wang, N. Komodakis, N. Paragios, Markov random field modeling, inference & learning in computer vision & image understanding: A survey**, Comput. Vis. Image Underst. 117 (2013) 1610–1627. <http://dx.doi.org/10.1016/j.cviu.2013.07.004>.
- [51] **N. Dalal, B. Triggs**, Histograms of oriented gradients for human detection, in: 2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., IEEE, San Diego, CA, USA, 2005, pp. 886–893. <http://dx.doi.org/10.1109/CVPR.2005.177>.
- [52] **Dong-Chen He, Li Wang**, Texture unit, texture spectrum and texture analysis, in: 12th Can. Symp. Remote Sens. Geosci. Remote Sens. Symp., IEEE, 1989, pp. 2769–2772. <http://dx.doi.org/10.1109/IGARSS.1989.575836>.
- [53] **L. Wang, D.-C. He**, Texture classification using texture spectrum, Pattern Recognit. 23 (1990) 905–910. [http://dx.doi.org/10.1016/0031-3203\(90\)90135-8](http://dx.doi.org/10.1016/0031-3203(90)90135-8).

- [54] **R. Funayama, H. Yanagihara, L. Van Gool, T. Tuytelaars, H. Bay**, Robust interest point detector and descriptor, US 8,165,401 B2, 2006. <https://www.google.com/patents/US8165401>.
- [55] **X. Wang, T.X. Han, S. Yan**, An HOG-LBP human detector with partial occlusion handling, in: 2009 IEEE 12th Int. Conf. Comput. Vis., IEEE, Kyoto, 2009, pp. 32–39. <http://dx.doi.org/10.1109/ICCV.2009.5459207>.
- [56] **D.G. Lowe**, Object recognition from local scale-invariant features, in: Proc. Seventh IEEE Int. Conf. Comput. Vis., Vol. 2, IEEE, 1999, pp. 1150–1157. <http://dx.doi.org/10.1109/ICCV.1999.790410>.
- [57] **D.R. Cox**, The regression analysis of binary sequences, J. R. Stat. Soc. 20 (1958) 215–242. <http://www.jstor.org/stable/2983890>.
- [58] **S.H. Walker, D.B. Duncan**, Estimation of the probability of an event as a function of several independent variables, Biometrika 54 (1967) 167. <http://dx.doi.org/10.2307/2333860>.
- [59] **Virginia Menezes, Vamsikrishna Patchava, M. Surya Deekshith Gupta**, "Surveillance and monitoring system using Raspberry Pi and SimpleCV", 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), vol. 00, no. pp. 1276-1278, 2015, doi:10.1109/ICGCIoT.2015.7380661.
- [60] Raspberry Pi, 2015. <https://www.raspberrypi.org>.(accessed may 4, 2015).
- [61] **Ying Wen Bai, Li-SihShen and Zong-Han Li**, IEEE 2010, Design and Implementation of an Embedded Home Surveillance System by Use of Multiple Ultrasonic Sensors.
- [62] Pir motion sensor Inc., How pirs work, 2014. <https://learn.adafruit.com> (accessed 28.01.14).
- [63] OpenCV, 2016. <http://opencv.org/>.(accessed December 23, 2016).
- [64] **P. Viola and M. Jones**, "Rapid object detection using a boosted classifier of simple features," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2001, vol. 1, pp. 511–518.
- [65] **Cordiner, A.; Ogunbona, P.; Wanqing Li**, "Face detection using generalised integral image features," Image Processing (ICIP), 2009 16th IEEE International Conference on, vol., no., pp.1229,1232, 7- 10 Nov. 2009.

- [66] **Xiaohu Ma, Shigang Chen, Shukui Zhang**, "AdaBoost Face Detection Based on Haar-Like Intensity Features and Multi-Threshold Features", *Multimedia and Signal Processing, International Conference on*, vol. 01, no., pp. 251-255, 2011, doi:10.1109/CMSP.2011.58.
- [67] **S. Z. Li, Daniel Gatica-Perez, Dong Zhang**, "Real-Time Face Detection Using Boosting in Hierarchical Feature Spaces", vol. 02, no, pp. 411-414, 2004, doi:10.1109/ICPR.2004.1334238.
- [68] **Samiksha Agrawal, Pallavi Khatri**, "Facial Expression Detection Techniques: Based on Viola and Jones Algorithm and Principal Component Analysis", vol. 00, no, pp. 108-112, 2015, doi:10.1109/ACCT.2015.32.
- [69] **M. Turk and A. Pentland**, "Eigenfaces for recognition", *Cognitive Neuroscience*, 3:72{86, 2001.
- [70] **S. Z. Li and A. K. Jain (eds.)**, "Handbook of Face Recognition" Springer Verlag, Secaucus, NJ, 2005.
- [71] **Phillips, P., Grother, P., Micheals, R.J., Blackburn, D.M., Tabassi, E., Bone, J.M.**: Face recognition vendor test 2002 results. Technical report (2003).
- [72] **N. Dalal**, Finding people in images and videos (Ph.D. thesis), INRIA Grenoble, 2006.
- [73] **P. Dollar, C. Wojek, B. Schiele, P. Perona**, Pedestrian detection: an evaluation of the state of the art, *Pattern Anal. Mach. Intell. IEEE Trans.* 34 (2012) 743–761.
- [74] **Satpathy, X. Jiang, H.-L. Eng**, Human detection by quadratic classification on subspace of extended histogram of gradients, *Image Process. IEEE Trans.* 23 (2014).



## CURRICULUM VITA

### 1-Personal Details:

|                        |                                                                          |
|------------------------|--------------------------------------------------------------------------|
| Full Name:             | Nashwan Adnan Othman Aziz                                                |
| Date / Place of birth: | 01-01-1992 Ninawa - Iraq                                                 |
| Address:               | BARDARASH - IRAQ                                                         |
| Gender:                | Male                                                                     |
| Marital Status:        | Single                                                                   |
| Nationality:           | Iraqi                                                                    |
| Email address:         | <a href="mailto:nashwan.adnan92@gmail.com">nashwan.adnan92@gmail.com</a> |

### 2-Education:

| Level | Majored in                                                                                      | Field of study              | Duration |      | Name of the university                                   |
|-------|-------------------------------------------------------------------------------------------------|-----------------------------|----------|------|----------------------------------------------------------|
|       |                                                                                                 |                             | From     | To   |                                                          |
| B.Sc. | Computer science<br>Rank (4) in a class of (59) student.<br>Average Grade %74.403<br>July 2013. | Computer science (Tech.Eng) | 2009     | 2013 | University Salahadeen–<br>Computer Science College/Erbil |

### 3-Skills:

|                                   | Program                                                                                                                                               | Proficiency    |
|-----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|----------------|
| <b>Graphics</b>                   | Adobe (Photoshop-primer-after effect ), movies maker ...etc.                                                                                          | <b>Good</b>    |
| <b>Languages</b>                  | C++, C#, Python, Visual Basic, ...etc.                                                                                                                | <b>Good</b>    |
| <b>Windows</b>                    | Windows 10, Windows 8, Windows Seven, Windows XP, Linux, ...etc.                                                                                      | <b>V. Good</b> |
| <b>Others</b>                     | Microsoft Office 2003,2007,2010,2013 , Matlab, ...etc.                                                                                                | <b>V. Good</b> |
| <b>E-marketing<br/>E-commerce</b> | Facebook, e-mail, YouTube, google group, yahoo, Hotmail, design website (word press, html, Css, php, java script, Dreamweaver) ,Gmail, twitter...etc. | <b>V. Good</b> |
| <b>Network</b>                    | Wireless network, cisco packet tracer, configuration...etc.                                                                                           | <b>V. Good</b> |
| <b>Database</b>                   | MySQL -Access-Sql server...etc.                                                                                                                       | <b>Good</b>    |

### 4-Language Skills:

| Foreign Language | Competence |           |           | proficiency |
|------------------|------------|-----------|-----------|-------------|
|                  | Reading    | Writing   | Speaking  |             |
| <b>English</b>   | V. Good    | V. Good   | Good      | V. Good     |
| <b>Arabic</b>    | Excellent  | Excellent | Excellent | Excellent   |
| <b>Kurdish</b>   | Excellent  | Excellent | Excellent | Excellent   |

**5-Work Experience:**

| <b>Company</b>                                                                                           | <b>Job</b>                                                                                                                                                                                                                                                                                      | <b>Year</b>  |
|----------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| Teachers                                                                                                 | Responsible for teaching students &&<br>Responsible for the installation of surveillance cameras, devices and early warning devices and safety and fire safety<br>responsible for the configuration, troubleshooting, repair, updating and maintaining workstations, PCs, Printers, and copiers | 2013-Present |
| Education                                                                                                | I opened a learning course, trained more than 100 school director for some software applications such as (AscTimeTables, Excel....)                                                                                                                                                             | 2014         |
| Cultural Center                                                                                          | Manager                                                                                                                                                                                                                                                                                         | 2013-2016    |
| Cultural Center                                                                                          | I opened a learning course about some IT Applications such as (Photoshop course, internet, Excel, Word...) for many groups of people.                                                                                                                                                           | 2013-2015    |
| The independe high electoral commission                                                                  | Data Entry                                                                                                                                                                                                                                                                                      | 2014-2015    |
| <b>Note/</b> I have able to Fast Learning for anything's especially IT Applications and I'm Hard Worker. |                                                                                                                                                                                                                                                                                                 |              |