A PERSONAL SYSTEM SECURITY WITH CONTINUOUS AUTHENTICATION

by

M. SAI SHUSHMA	**411545**
A. BHAVYA PRIYA	**411509**
T. MANIKANTA MALLIKARJUNAM	**411570**

Under the guidance of

Dr. A. GOUTHAM REDDY

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY ANDHRA PRADESH

TADEPALLIGUDEM-534102, INDIA

MAY-2019

PERSONAL SYSTEM SECURITY WITH

CONTINUOUS AUTHENTICATION

Thesis submitted to
National Institute of Technology Andhra Pradesh
for the award of the degree

of

Bachelor of Technology

by

M. SAI SHUSHMA	**411545**
A. BHAVYA PRIYA	**411509**
T. MANIKANTA MALLIKARJUNAM	**411570**

Under the guidance of

Dr. A. GOUTHAM REDDY

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY ANDHRA PRADESH

TADEPALLIGUDEM-534102, INDIA

MAY-2019

DECLARATION

I declare that this written submission represents my ideas in my own words and where others ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

……..	……..	……..
M. SAI SHUSHMA	A. BHAVYA PRIYA	T. MANIKANTA
411545	411509	411570
Date: 13-05-2019	Date: 13-05-2019	Date: 13-05-2019

CERTIFICATE

It is certified that the work contained in the thesis titled "**A Personal system security using Continuous Authentication**" by "T. Manikanta Mallikarjunam, bearing Roll No: 411570" has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Dr. A. Goutham Reddy
Computer Science and Engineering
N.I.T. Andhra Pradesh
May, 2019

Acknowledgement

I take this opportunity to express a deep sense of gratitude and indebtedness to my project guide Dr. A.Goutham Reddy, Assistant Professor, Department of Computer Science and Engineering, National Institute of Technology, Andhra Pradesh for providing guidance, encouragement and inspiration throughout the project work. Without his invaluable guidance, this work would never have been a successful one.

I am thankful to all my friends, classmates for their constructive ideas and suggestions. I want to express sincere regards to my parents for their blessings, love and constant support throughout my work. Finally, I would like to express my deepest appreciation to all those who helped me in completing this thesis.

Maturi Sai Shushma

TABLE OF CONTENTS

List of figures

Abstract

Motivated primarily by the need for an unobtrusive, continuously monitoring system. Most users are familiar with the different forms of verification needed to login to their computers, access their email accounts or open their company's shared server. But in most of these instances, the user authenticates once, leaving these systems vulnerable to security breaches for the remainder of the session. Based on this, we put forward a method named as de-authentication that detects the absence of a previously - authenticated user and immediately terminates that user's active session. This would allow the application to distinguish among the legitimate users and provide them access to the system accordingly. This might be something as simple as recording a log of actions or might be enforcement of privilege restrictions. For any system user's comfort is important along with security. So we also propose Re-authentication mechanism which detects the previously authenticated user.

Contents

1. INTRODUCTION

Today, nearly all of an agency's mission-critical functions depend on safe and secure information technology systems. With cyber threats ever evolving and growing at an exponential rate, and increased reliance on technology to deliver core services, a robust defense is needed. Continuous Monitoring is certainly a best practice for developing a secure system. The following part of the introduction describes the terminologies and technologies we used for implementing our proposed approach.

1.1 Authentication

Authentication is the process of determining whether someone or something is, in fact, who or what it declares itself to be. Authentication technology provides access control for systems by checking to see if a user's credentials match the credentials in a database of authorized users or in a data authentication server. Users are usually identified with a user ID, and authentication is accomplished when the user provides a credential, for example, a password, that matches with that user ID.

Authentication Factors

Authenticating a user with a user ID and a password is usually considered the most basic type of authentication, and it depends on the user knowing two pieces of information: the user ID or username, and the password. Since this type of authentication relies on just one authentication factor, it is a type of single-factor authentication.

Two-factor authentication: Two-factor authentication adds an extra layer of protection to the process of authentication. 2FA requires that a user provide a second authentication factor in addition to the password. 2FA systems often require the user to enter a verification code received via text message on a pre-registered mobile phone, or a code generated by an authentication application.

Multi Factor authentication: Multi Factor authentication requires users to authenticate with more than one authentication factor, including a biometric factor like a fingerprint or facial recognition, a possession factor like a security key fob or a token generated by an authenticator app.

One-time password: A one-time password is an automatically generated numeric or alphanumeric string of characters that authenticates a user. This password is only valid for one login session or

transaction and is usually used for new users, or for users who lost their passwords and are given a one-time password to login and change to a new password.

Three-factor authentication: Three-factor authentication (3FA) is a type of MFA that uses three authentication factors, usually a knowledge factor (password) combined with a possession factor (security token) and the inherence factor (biometric).

Biometrics: While some authentication systems can depend solely on biometric identification, biometrics are usually used as a second or third authentication factor. The more common types of biometric authentication available include fingerprint scans, facial or retina scans, and voice recognition.

Mobile authentication: Mobile authentication is the process of verifying user via their devices or verifying the devices themselves. This lets users log into secure locations and resources from anywhere. The mobile authentication process involves multi factor authentication that can include one-time passwords, biometric authentication or QR code validation.

Continuous authentication: With continuous authentication, instead of a user is either logged in or out, a company's application continually computes an "authentication score" that measures how sure it is that the account owner is the individual who's using the device.

API authentication: The standard methods of managing API authentication are HTTP basic authentication; API keys and OAuth.

In HTTP basic authentication, the server requests authentication information, i.e., a username and password, from a client. The client then passes the authentication information to the server in an authorization header.

In the API key authentication method, a first-time user is assigned a unique generated value that indicates that the user is known. Then each time the user tries to enter the system again, his unique key is used to verify that he is the same user who entered the system previously.

Open Authentication (OAuth) is an open standard for token-based authentication and authorization on the internet. OAuth allows a user's account information to be used by third-party services, such as Facebook, without exposing the user's password. OAuth acts as an intermediary on behalf of the user, providing the service with an access token that authorizes specific account information to be shared.

1.2 De-authentication

De-authentication in a simple sense means the authenticated user gets disconnected with the authenticated system. In general when the authenticated system feels the user is not the one who initially authenticated then the system stop service to the user. In order to de-authenticate the user systems have to monitor the user continuously. In order to monitor the user continuously there are some techniques used by the system. Some systems use keystroke dynamics as a parameter and decide whether the user is legitimate or not. Some systems use the webcam and monitor the user face continuously. If the user is not found in the picture taken by the webcam then the gets de-authenticated.

De-authentication is important because we are defining a system is secured only if the services provided by the system to the legitimate user. So many systems today providing good authentication systems. But de-authentication is also as important as authentication because if the system feels the user is not the correct one then the system should not provide service to the user. Then only complete security of the system is ensured.

1.3 Re-authentication:

In general, authentication is a process which ensures high security in confirming the user. In order to ensure a high-security system takes secured data of the user. Sending the same secured data of the user to the system causes a lot of discomfort and security issues. So in order to solve this problem system should use re-authentication which should be simple than the authentication.

After de-authentication, if the system should found the legitimate user it should continue the services to the user. In order to confirm the user again, the system should authenticate to the user in a different manner such that it does not provide any discomfort to the user.

For any system to provide high security and comfort it should provide Authentication, De-authentication, and Re-authentication. First, the system should authenticate to the system using high secured data. During continuous monitoring, if the user initially authenticated is found absent then the system should de-authenticate. If the user is found again then the system should confirm the user using re-authentication mechanism. So the three mechanisms help in ensuring the complete system security.

1.4 D6T SENSOR:

Automation of devices has revolutionized the comfort levels of human life. New devices are equipped with automated systems for reducing the need for human effort. The eco-friendly automation is relying on sensors, transducers, and control systems. Therefore researchers have focused much on developing sensors with new technologies for sensing and detection of the

targeted objects. Human exhibits various unique characteristic like body temperature, shape and color, and release of CO2.

Our work aims to create a system which can respond to the human body temperature for detection. IR sensors are best and accurate to measure radiated heat by human body. It is the most precise characteristic for human detection in sensitive locations. It is well known that anything that has a temperature above absolute zero emits infrared radiation. Another crucial challenge is human body temperature is not constant which varies from 36.5 to 37.5 ℃ due to wearable clothes, external conditions like seasons and day and night timings. The primary factors for localization of human presence may include the distance between the heat source and sensor system. The body surface radiation is absorbed by the sensor can range from 20℃ to 40℃ depending on the external conditions. For indoor applications, the sensor should be able to differentiate human from static objects/things like metal objects, wooden furniture, and plastic objects even though the human is mobile or immobile. The parameters like distance, angle and coordinates are calculated using the position of the sensing device for accurate human location in the field of view (FOV) or targeted location. Therefore, it is highly essential to choose the IR sensors for concerning range, accuracy, and FOV. This research work has proposed Omron D6T MEMS IR sensor which results in 4x4 array pixel low-resolution thermal image as graphical output. The outcome of this work is to visualize the thermal signature of human and human-object interaction in two dimension (2D) using thermopile modules of Omron D6T sensor. The position of sensors is also another crucial parameter to consider for better coverage of FOV. The number of required sensors is based on size of FOV and human count by avoid the interference range between multiple sensors.

Unlike CCTV cameras, the low resolution 2D thermal signature output of the thermopile sensor is not smooth to interpret thereby reduces the privacy concerns of the users. The 2D array matrix output of Omron D6T MEMS series sensor identifies the pixels of relatively high temperature than the surrounding and gives the location and presence of the human in range. The PIR sensors were most commonly preferred to detect human presence before the thermal sensors due to low cost. However, the PIR sensor can able to identify the signals from the person in motion and simultaneously it results in the false negative when the person was stationary. PIR sensor also has another limitation of time transmission delay of approximately 2 seconds to 9 minutes. On the other hand, the thermal sensor can continuously differentiate a stationary human from other static or dynamic objects.

Omron Electronics D6T Series MEMS Thermal Sensors are super-sensitive infrared temperature sensors that make full use of Omron's proprietary MEMS sensing technology. Unlike

typical pyroelectric human presence sensors that rely on motion detection, the D6T thermal sensor is able to detect the presence of stationary humans by detecting body heat, and can therefore, be used to automatically switch off unnecessary lighting, air conditioning, etc. when people are not present. As the D6T sensor is also able to monitor the temperature of a room, it can also be used to continually maintain optimal room temperature levels, instantly sense unusual changes in temperature thereby detecting factory line stoppages, or discover areas of overheating for early prevention of fire outbreaks.

D6T Features:

- Achieves world's highest level of SNR.
- Accurate temperature measurements with little impact from outside.
- Superior noise immunity with a digital output
- High-precision area temperature detection with low crosstalk field of view characteristics
- RoHS compliant
- Compact size for space savings and embedded applications
- Converts sensor signals to digital temperature output, allowing easy use of microcontroller

Structure:

The D6T series of MEMS Thermal Sensors consists of a small circuit board onto which a silicon lens, thermopile sensor, specialized analog circuit, and logic circuit for conversion to a digital temperature value are arranged. This product only requires one connector to connect these modules.

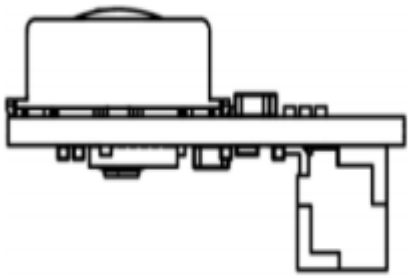

Fig. 1 D6T structural view

Fig. 2 D6T sensors

External Dimensions:

This product features a circuit board size of 14 mm x 18 mm. Even more compact size of 11.6 mm x 12 mm is also available. Refer to the product catalog for more information on mounting

areas and positioning of the circuit board. Refer to Chapter 6 for more information on compatible connectors.

Principles of Operation:

The following list describes an overview of the measuring operation of the MEMS Thermal Sensors. The silicon lens focuses radiant heat (far-infrared rays) emitted from objects onto the thermopile sensor in the module.

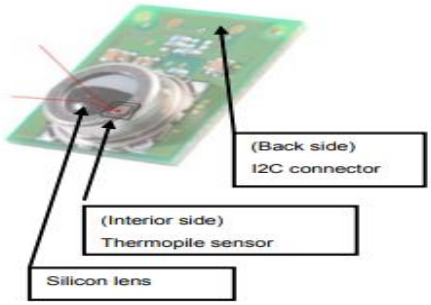

Fig.3 Components of D6T

The thermopile sensor generates electromotive force in accordance with the radiant energy (far-infrared rays) focused on it. The values of this electromotive force and the internal thermal sensor are measured. Then, the device calculates the measured value (temperature of the object) via an interpolation calculation that compares the measured values with an internally stored lookup table. The measured value is output via the I2C bus and read using a host system.

Product Features:

MEMS Thermal Sensors measure the surface temperature of objects. The D6T-44L-06 model features 16 channels in a 4 x 4 arrangement. The D6T-8L-09 features a single 8-channel array. The D6T-1A-01/-02 models feature a 1-channel sensor chip. The module has been optimized by placing the specialized downstream processing circuit adjacent to the sensor chip to achieve low-noise temperature measurements.

Using our MEMS Thermal Sensors as a human sensor eliminates the problems in using conventional pyroelectric sensors to detect the presence of people. Pyroelectric sensors can be used to detect movement of people based on the principle of detecting change components of infrared rays, but the measurement signal is lost during times of no movement. Conversely, Thermal Sensors continue to generate a measurement signal during times of no movement.

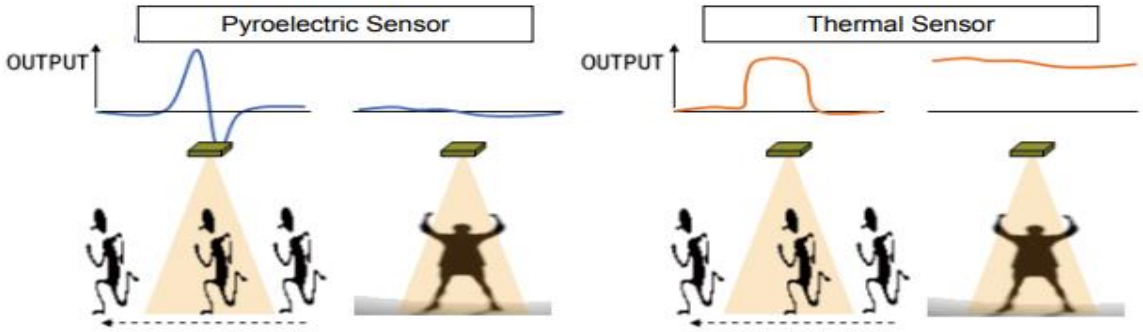

Fig.4 Output of Pyroelectric sensor Fig.5 Output of Thermal sensor

MEMS Thermal Sensors feature a silicon lens optically designed to have specific sensitivity characteristics. Our Thermal Sensors feature the same field of view (FOV) at a maximum sensitivity of 50% as general sensors.

The sensitive areas of elements are wider than the FOV-specification width. If the size of the measured object is smaller than the sensitive area of an element, the background temperature of objects other than the intended object will become a factor. Our Thermal Sensors use a reference heat source (a blackbody furnace) to correct temperature values. However, note that differences in emissivity due to the composition of measured objects, surface shape, and the occupancy ratio of objects within sensitive areas all affect temperature values.

The measurable area (FOV) enlarges as the distance between the measured object increases. The occupancy ratio of objects (people) in the FOV reduces as the distance increases. For this reason, as the distance increases, the temperature values become more a representation (level of influence) of the background temperature than the temperature of the intended object (people). In other words, to correctly measure the temperature of the intended objects, the measured object must be sufficiently larger than the FOV area.

Using a MEMS Thermal Sensor as a human sensor is limited to close-distance applications for simple determination of temperature value only. To increase the detection distance, determination accuracy must be improved through software processing that factors temporal changes, the position of heat sources, human behavior information, and so on.

Usage requirements:

Connections Pins:

To connect the sensor to the circuit, pins of the sensor are required. Pin diagram of the sensor is given below.

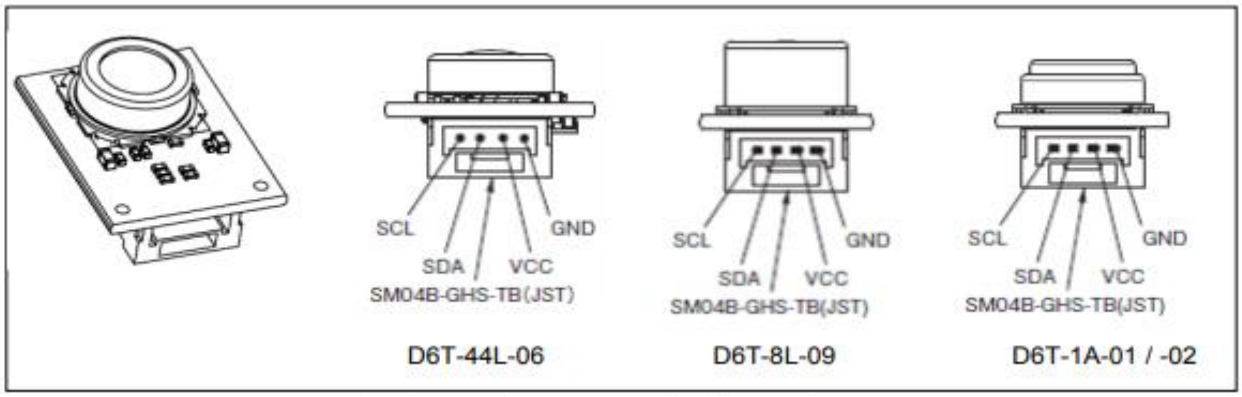

Fig.6. Pin connections

Electrical connections:

The basic connections of the D6T sensor with Microcontroller is shown in the diagram

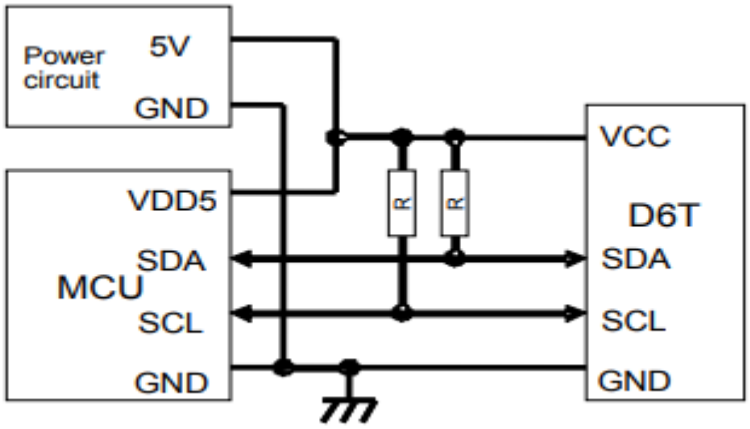

Fig.7.Electical Connections

Product Specifications:

D6T sensor is very sensitive, so it should be operated in under given specifications of the sensor by the manufacturer. Different properties like temperature, voltage, current, humidity, data rate are shown clearly in the table below.

Item	Specification
Storage temperature range [°C]	-10 to 60° C (with no ice or no dew condensation)
Operating temperature range [°C]	0 to 50° C (with no ice or no dew condensation) * Guaranteed accuracy, see the figure on the next page.
Storage humidity range [%RH]	85%RH or less (with no dew condensation)
Operating humidity range [%RH]	20 to 85%RH (with no dew condensation)
Supply voltage [V]	4.5 to 5.5
Maximum output voltage [V]	0.8Vcc to Vcc
Minimum output voltage [V]	0 to 0.2Vcc
Current consumption [mA]	Typ.5
Output	Temperature values
Object temperature accuracy [°C]	±1.5
Digital interface	I2C (Synchronous serial communication)
Data update rate	Max.250ms

Table.1

Applications

- Energy Management: Detecting the human body for energy savings and comfort
- Security: Detecting the human body in a dark area
- TV/PC: Detecting the human body for screen saving
- Microwave Oven: Temperature control for cooking foods
- Refrigerator: Cooling warm food rapidly to preserve freshness
- Air Conditioner: Temperature control for floor temperature
- Cooking heater: Detecting the pan temperature for automatic cooker
- Temperature sensing: Detecting unusual conditions (overheating)
- Factory automation

1.5 Facial Recognition

Facial recognition is a biometric software application capable of uniquely identifying or verifying a person by comparing and analyzing patterns based on the person's facial contours. Facial recognition is mostly used for security purposes, though there is increasing interest in other areas of use. In fact, facial recognition technology has received significant attention as it has the

potential for a wide range of application related to law enforcement as well as other enterprises. Facial recognition is also known as face recognition.

There are different facial recognition techniques in use, such as the generalized matching face detection method and the adaptive regional blend matching method. Most facial recognition systems function based on the different nodal points on a human face. The values measured against the variable associated with points of a person's face help in uniquely identifying or verifying the person. With this technique, applications can use data captured from faces and can accurately and quickly identify target individuals. Facial recognition techniques are quickly evolving with new approaches such as 3-D modeling, helping to overcome issues with existing techniques.

How it works:

The mathematical algorithms of biometric facial recognition follow several stages of image processing:

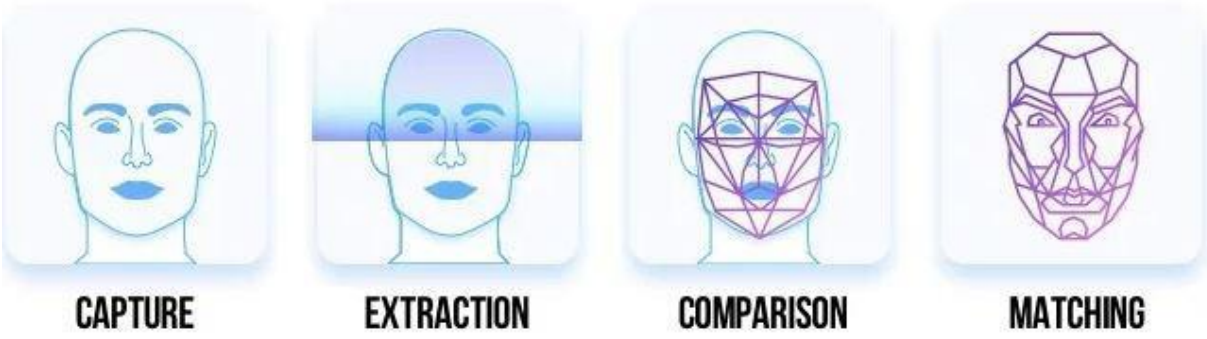

Fig.8. Stages of Image Processing

- **Capture**

  The first step is for the system to collect physical or behavioral samples in predetermined conditions and during a stated period of time.

- **Extraction**

  Then, all this gathered data should be extracted from the samples to create templates based on them.

- **Comparison**

  After the extraction, collected data is compared with the existing templates.

- **Matching**

  The final stage of face detection technology is to make a decision whether the face's features of a new sample are matching with the one from a facial database or not. It usually takes just seconds.

There are many ways to face recognition. Here we use OpenCV for face recognition. In face recognition, the image first prepared for preprocessing and then trained the face recognizer to recognize the faces. After teaching the recognizer, we test the recognizer to see the results. The OpenCV face recognizer are of three types, which are as follows-

1.5.1 EigenFaces Face Recognizer

EigenFaces face recognizer views at all the training images of all the characters as a complex and try to deduce the components. These components are necessary and helpful (the parts that grab the most variance/change) and discard the rest of the images, This way it not only extracts the essential elements from the training data but also saves memory by rejecting the less critical segments.

1.5.2 FisherFaces Recognizer

Fisherfaces algorithm, instead of obtaining useful features that represent all the faces of all the persons, it removes valuable features that discriminate one person from the others. This features of one person do not dominate over the others, and you have the features that distinguish one person from the others.

1.5.3 Local Binary Patterns Histograms

We know that Eigenfaces and Fisherfaces are both affected by light and in real life; we cannot guarantee perfect light conditions. LBPH face recognizer is an improvement to overcome this drawback. The idea is not to find the local features of an image. LBPH algorithm tries to find the local structure of an image, and it does that by comparing each pixel with its neighbouring pixels.

Applications:

- Payments
- Access and Security
- Criminal Identification

- Advertising
- Healthcare

Pros:

- The improvement of security level
- Easy integration process
- High Accuracy rates
- Full Automation

Cons:

- Processing and Storing
- Image size and Quality
- Surveillance Angle

1.6 Speech Recognition

Speech recognition is the ability of a machine or program to identify words and phrases in spoken language and convert them to a machine-readable format. Rudimentary speech recognition software has a limited vocabulary of words and phrases, and it may only identify these if they are spoken very clearly. More sophisticated software has the ability to accept natural speech.

It is a technology that allows spoken input into systems. You talk to your computer, phone, or device and it uses what you said as input to trigger some action. The technology is being used to replace other methods of input like typing, clicking, or selecting in other ways. It is a means to make devices and software more user-friendly and to increase productivity.

How it works

Speech recognition works using algorithms through acoustic and language modeling. Acoustic modeling represents the relationship between linguistic units of speech and audio signals; language modeling matches sounds with word sequences to help distinguish between words that sound similar. Often, hidden Markov models are used as well to recognize temporal patterns in speech to improve accuracy within the system.

Applications

The most frequent applications of speech recognition within the enterprise include call routing, speech-to-text processing, voice dialing, and voice search.

Pros and Cons

While convenient, speech recognition technology still has a few issues to work through, as it is continuously developed. The pros of speech recognition software are it is easy to use and readily available. Speech recognition software is now frequently installed in computers and mobile devices, allowing for easy access.

The downside of speech recognition includes its inability to capture words due to variations of pronunciation, its lack of support for most languages outside of English and its inability to sort through background noise. These factors can lead to inaccuracies.

Performance

Speech recognition performance is measured by accuracy and speed. Accuracy is measured with the word error rate. WER works at the word level and identifies inaccuracies in transcription, although it cannot identify how the error occurred. Speed is measured with the real-time factor. A variety of factors can affect computer speech recognition performance, including pronunciation, accent, pitch, volume and background noise.

It is important to note the terms speech recognition and voice recognition are sometimes used interchangeably. However, the two terms mean different things. Speech recognition is used to identify words in spoken language. Voice recognition is a biometric technology used to identify a particular individual's voice or for speaker identification.

2. MOTIVATION

In this world of modern technology, we follow different technologies for specific tasks in our daily life. Moreover, technology has been running in our daily lives in all sectors, no matter what industry you are dealing with, it has an impact in an unquestionable manner. Today each and every member, regardless of age are aware of the technical devices. Since last many years, technology has taken the world in terms of the products we purchase, communicate, the way we live, learn and has brought many lots of changes with this constant advancement of technology. Ensuring security for all these is becoming a necessary task. Whenever we thought of security, three words come to our mind i.e., confidentiality, integrity, and authentication. To ensure them appropriate steps have to be taken priorly.

Generally, system provides different services to the user. Before providing services, system checks whether it is providing to the legitimate user or not and also whether that services can be provided to the user. The first process is nothing but authentication and the second one is authorization. To provide secure authentication many companies like Google, Facebook, etc. have been developing many applications. Few techniques like asking user for credentials (username and password), two-factor authentication, multi-factor authentication, biometrics such as fingerprint, facial recognition, iris, palm vein, keystroke dynamics were developed and other new techniques are being developed for providing authentication in a more secured manner to avoid attacks.

The above-mentioned technologies take care whether the authentication is ensured or not at session start. The cases in which user authenticates with the system and then left the system for some time because of some other work, meanwhile intruder accessing the services are not addressed in the proposed approaches. The need for continuous monitoring of the user after successful authentication came into existence. Many systems are developed to address this issue but failed because of some reasons which were explained in the literature review. We proposed a new architecture which overcomes the drawbacks of proposed systems and provides authentication with continuous monitoring.

3.LITERATURE REVIEW

3.1 Continuous Authentication using Key Stroke & Mouse

The username-password based access control mechanism is a widely implemented security measure to protect the device from unauthorized access. This type of access control is generally implemented as a one-time proof of identity during the initial log on procedure (i.e. Static Authentication (SA)). The legitimacy of the user is assumed to be the same during the full session. Unfortunately, if the device is left unlocked and unattended, any person can have access to the same information as the genuine user. On the other hand, we have Continuous Authentication (CA), where the genuineness of a user is continuously monitored based on the biometric signature left on the device. When doubt arises about the genuineness of the user, the system can lock, and the user has to revert to SA. CA is not an alternative security solution for SA; it provides an added security measure alongside SA. In case the CA mechanism detects an impostor, the system should lock to avoid any damage done by, or information revealed to that impostor. The obvious requirements are to detect an impostor as fast as possible to control the amount of damage, while at the same time avoiding, to the largest possible extend, the incorrect locking out of the genuine user. Furthermore, should a CA mechanism, much more than a SA method, perform its tasks unnoticed to the user. This immediately rules out the use of knowledge or possession based authentication mechanism. Knowledge based systems will disturb the user when having to type a password, while possession based systems are not effective for users that do not remove their token when leaving the system unattended. Besides, a stolen token would give an attacker the same access rights as the genuine user and would not lead to detection by the computer system. This then leaves biometrics as a potential solution for continuous authentication. In the proposed system the behavior of the current user is compared to the normal behavior of the genuine user and deviation from this normal behavior will lead to a lockout. The motivation behind the use of behavioral biometrics is the unobtrusive nature of the data collection for some behavioral biometrics e.g. Keystroke Dynamics (KD) and Mouse Dynamics (MD). In this paper, we are not only looking at Continuous Authentication (CA) where the system checks if the current user is the genuine user, but also at Continuous Identification (CI) where the system tries to identify the current user of a system when the CA has detected that the current user is not the genuine user. CI can be used as a forensics tool. To the best of our knowledge is this the first time that the CI issue is raised in research. Performing CA-CI by analyzing the user's behavior profile is challenging due to the limited amount of information that is available and the large intra-class variations. To mitigate the classification challenges we also developed Pairwise User Coupling (PUC) for CI.

The summary of the paper is as follows:

- Introduce the concept of Continuous Authentication and Identification, that provides the combination of security and forensics;
- Develop a novel identification algorithm with Pairwise User Coupling that could be applied to other pattern identification problems.
- A combination of KD and MD is applied to avoid a situation where an attacker avoids detection by restricting as much as possible to one input device because the system only checks the other input device. Combination of these two modalities can improve the overall system performance. It is also believed that it is very difficult to spoof more than one behavioral biometric modality simultaneously.

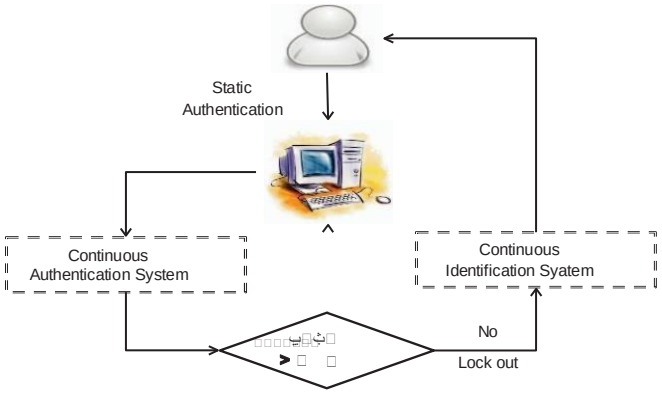

Fig.9. Flow chart for CA and CI in a system

Person authentication and identification by analyzing the user's behavior profile is challenging due to limited information, large intra-class variations and the sparse nature of the information. It is stated that statistical analysis (i.e. distance based classifiers) failed to achieve the desired results due to these challenges. Therefore, four different machine learning approach was used in the research, more precisely we have used Decision Tree (DT), Counter Propagation Artificial Neural Network (CPANN), ANN and SVM classifiers in the research [2, 3, 6, 10].

Dataset Description

Despite a high degree of privacy concern, it had got 25 volunteers to participate in the experiment. The volunteers installed the data logging software and collected data continuously for 5 to 7 days. All the participants of this data collection process are university students and staff members and they are regular computer users. In this case will the user be focused more on completing the task, and will their behavior not represent their normal behavior [7, 13]. For that reason is it not possible to easily extend the results from experiments in a controlled setting to predicted results in an uncontrolled or real world setting. To address this issue is our data collected

in an uncontrolled environment, where no instruction or any specific task was given to the user and all of our participants used their own system to remove the hardware changes effect on the natural behavioral pattern. We split the data of each participant into three non-overlapping parts. For each user, the Training Set (M) used to build and train the system consists of 35% of the data from that user. Similarly contains the Validation Set (V), used for parameter adjustment of the algorithms used in this research, 10% of the data, and the Test Set (T), used to test the system performance, contains the remaining 55% of the data.

Feature Extraction

Keystroke Features

Every keystroke event k is encoded as k =(A,Tp,Tr), where Tp,Tr are the timestamps in milliseconds for key press and key release and A is the value of the pressed key. The raw keystroke events are converted into two different actions.

1. Single Key Action: The feature vector $FVsk\ i$ = (Ai, i), where Ai is the ith key, and di is the duration of the ith pressed key (i.e. $di = Tr\ i -$). 2. Key Digraph Action: The feature vector $FVdi\ i$ = (ti, ,$rr\ i$,$lpp\ i$), where ti is the total time duration of two consecutive keystrokes (i.e. $ti = Tr\ i$+1−$Tp\ i$). Furthermore do , $lrr\ i$, and $lpp\ i$ represent latencies between the ith and (i + 1)th keys, in particular $lrp\ i = Tp\ i$+1 −$Tr\ i$, $lrr\ i = Tr\ i$+1 −$Tr\ i$, and $lpp\ i = Tp\ i$+1 −$Tp\ i$. Two consecutive keystrokes are considered to be a Key Digraph Action when $lrp\ i$ < 2000ms.

Mouse Features

The raw mouse events are converted into four different actions.

1. Mouse Single Click Action: The feature is similar to a Single Key Action, i.e. the time duration between mouse button press and release.

2. Mouse Double Click Action: The features are the same as those of a Key Digraph Action. Two consecutive mouse clicks are considered to be a double click when $lrp\ i$ < 1000ms.

3. Mouse Move Action: This action was formed by the sequence of mouse move events.

4. Mouse Drag-Drop Action: This action is very similar to the Mouse Move Action, but for this action first there has to be a mouse click down event followed by mouse move sequences and then mouse click up event. We computed Mouse Move Action and Mouse Drag-Drop Action features according to [12].

Before building the classifier models the feature selection technique is applied only to the mouse move and drag-drop action. This process was followed for both the CA and CI classifier models.

This feature selection process is based on maximization of the separation (i.e.Kolmogorov–Smirnov test between two Multi-variate Cumulative Distributions (MVCDF)).

3.2 Continuous Authentication using Key Stroke & Face

This includes the usage of the two different biometric recognition processes and their effect on the decision making of the system, as depicted in Figure 10.

TV stands for the trust value. As shown in Figure 10, the system additionally uses a timer and two thresholds:

- Timer t1: This timer measures the constant time between camera captures.
- Threshold 1 (TH1): If the trust value falls below this threshold, the user will be logged off.
- Threshold 2 (TH2): If the trust value falls below threshold 2, but is still higher than threshold 1, additional face recognition is performed outside of the timer cycle.

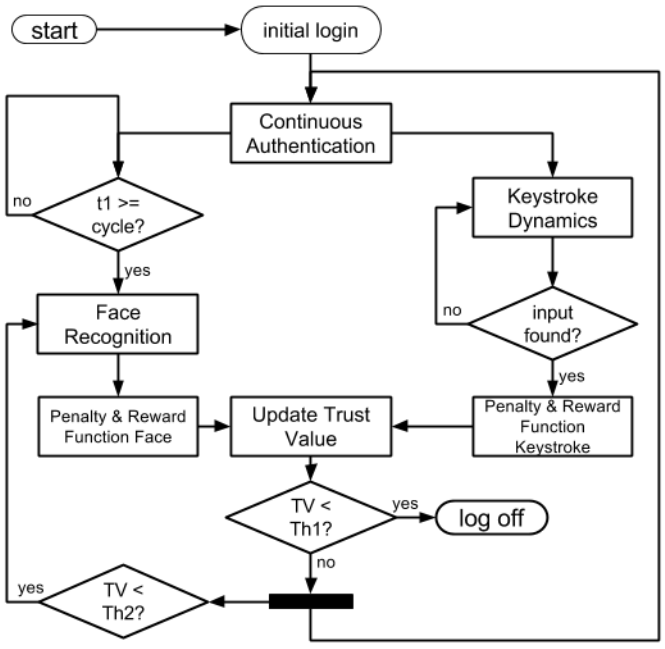

Fig.10. Flow chart for continuous authentication with facial recognition and key stroke dynamics

The key element of the system is the trust value. This value measures the confidence that the current user of the system is a genuine user. Certain actions will have an impact on the trust value, as they will increase or decrease the trust value. If the trust value remains high, the user is believed to be genuine and can continue his/her work without interruption. If the trust value falls

below a certain threshold (threshold TH1), the user is believed to be an impostor and is logged off. The trust value is determined as a function of the performances of the keystroke and face matchers, their resulting comparison scores, and the timer.

The Keystroke dynamics is measured continuously. Each input is transformed and compared to the biometric reference saved in the database. The resulting comparison score is measured as the distance between the input and the reference and can have a big impact on the trust value, meaning that a high distance will decrease the trust value and a low distance will increase the trust value, depending on the deviation from the genuine behavior.

Face recognition is performed periodically. Using face recognition only periodically lowers the negative impact on the user's privacy and makes it harder to perform activity recognition compared to permanent video surveillance. If the trust value falls below a threshold (threshold $TH2$), an additional attempt at face recognition is performed outside this periodic cycle. This gives an additional opportunity to increase the trust value of a genuine user, preventing a false rejection in the cases where the trust value is decreased due to outliers. It also speeds up the rejection of impostors.

The effect of the face recognition on the trust value is handled in a similar way to the keystroke dynamics, thus the deviation from the threshold for a match has an impact on the loss of trust in the case of a mismatch. If no face is found in the capture, the trust value is decreased by a fixed value.

1) Genuine User Scenarios:

- The user works normally on the PC. Keystroke dynamics are measured continuously and face recognition periodically as described above. Since there are deviations in the behavior of the user, the usage of both biometric characteristic should prevent or limit false rejections of genuine user despite these deviations.
- The user is logged in but leaves the PC. The periodic face recognition won't find a face in the image, thus the trust value will decrease. Since there are no keystrokes, the trust value won't be able to increase. This will trigger the first threshold and therefore the additional attempt at face recognition. This attempt will fail too and the user is soon logged off, as the trust value falls below threshold $TH1$.
- The user works on the PC, but moves around a lot. This will possibly result in a

failure to perform the periodic face recognition and thus a degradation of the trust value. The keystroke dynamics should keep the trust value above threshold $TH1$. If the periodic face recognition happens during a phase of inactivity and threshold $TH2$ is reached, a second attempt at face recognition is performed.

2) Impostor User Scenarios:

- An impostor is working on the PC. Keystroke dynamics are measured continuously. Since the deviation of the behavior of the genuine user is bound to be quite high, the trust value will decrease fast. Face recognition won't be able to verify the user either, so the trust value will degrade even faster.
- An impostor is working on the PC, but moved out of sight of the sensor performing the face recognition. Since no face is found, the trust value will decrease. This will result in the same scenario as impostor scenario 1. This scenario requires the attacker to notice or to know about the sensor for face recognition.
- An impostor is working on the PC, but moved out of sight of the sensor for the face recognition and is only using the mouse. Again face recognition won't be able to find a face and thus lower the trust value until the user is logged out. This scenario requires the attacker to notice or to know about the sensor for face recognition and the knowledge that keystrokes are used to verify the identity of the user.
- An impostor is working on the PC by only using the mouse (or as little keystrokes as possible) and has acquired a mask, or similar presentation attack tool to fool the face recognition. In order to remedy this presentation attack detection should be added to the face recognition subsystem. Otherwise this might be a scenario in which the intruder might gain access to the system for a longer period of time depending on the quality of the face recognition versus the quality of the mask.
- In order to conduct a sufficient evaluation of the proposed system and the used algorithms, the biometric data of each participant was divided into three parts. The whole biometric data of a participant is from now on referred to as a data set. A subset of a data is a block.
- The partition was done as follows: The amount of images belonging to a data set was divided by three, this corresponds to the capture time. Each block contains one third of the face images. The boundaries of these blocks were then adjusted to coincide with recording sessions. That means, the boundaries were adjusted to match with the nearest start or end of recording session respectively. The result are three blocks of face images, where the

start and end of each block coincides with the start and end of a recording session. This was done to ensure that parts of a recording session would not be part of two different blocks. The timestamps of the start and end of each block of face images are then used to separate the keystroke data in blocks. As a result, the time-frame of a block of keystroke data is the same as the time-frame of the correlating block of face images. This resulted in three blocks of data for each data set, where the blocks are separated by the start and end of recording sessions.

For face recognition, an image from the first five images of each block was selected as a reference for subject. The selection of the picture was based on pose and overall quality. One of the two remaining blocks was then used for the genuine test. Each block was selected as a reference once, and then compared to another block of the same data set as a genuine test. Therefore, three genuine tests were performed per data set. After that, each block of the other data sets was compared to that reference as an impostor test. This made it possible to conduct a total of
14 3 = 42 genuine tests and 42 39 = 1638 impostor tests. * *

B. Performance Metrics

In order to evaluate the performance of the algorithms used in this work, following metrics where used:

- ***Impostor Detection Rate (IDR):*** The IDR states rate of the successfully detected imposters.
- ***Average Number of False Rejections (ANFR):*** The ANFR indicates how many times the genuine user was falsely logged out during his session.
- ***Average Number of Genuine Actions (ANGA):*** The ANGA records how many action a genuine user was able to perform on average before being falsely rejected by the system.
- ***Average Number of Impostor Actions (ANIA):*** The ANIA describes how many actions an impostor was able to perform before being detected and logged off by the system. Note that only the number of actions leading to the first detection and log off were recorded.

For keystroke dynamics, an action is a keystroke. For face recognition, an action is a face recognition comparison on an image taken by web-cam. Consequently, it is easy to calculate the time the system needs to detect an impostor or the interval of false

rejections from the ANIA/ANGA of the face recognition system. In order to compare these metrics more clearly, the average of the IDR, ANFR, ANIA and ANGA over the users were calculated.

3.3 Assentication: PoPa biometric

PoPa posture pattern biometric works by monitoring, over time, changing pressure patterns exerted by a user seated in a typical office chair. PoPa relies on a combination of user behavioral patterns and physical characteristics. The latter includes: hip width, spine length, leg length, torso width, as well as overall weight. In addition, overall pressure distribution and its shifts are determined by the user's exact posture. Over time, it changes in a way that is unique to each user and that user's emotive state. This behavioral characteristic is also factored into PoPa.

A. Strengths & Weaknesses

Since exact distribution of seated pressure depends on the user's physical dimensions as well as on adopted postures, PoPa is a hybrid biometric blending physiological and behavioral factors. This allows it to benefit from some strengths of both. In particular, one's posture pattern can be captured in a strictly passive manner. Even though this property is shared by other biometrics, such as facial recognition or pulse response, posture pattern is not easily circumventable (unlike, e.g., facial recognition), and does not alter normal

However, posture pattern also inherits some weakness of both types of biometrics. As in any physiological biometric, it impossible to capture one's template (i.e., posture pattern) without the use of hardware specifically instrumented for this task. Fortunately, Po-Pa requires very little in terms of specialized hardware. As discussed later in Section VII, we constructed a PoPa prototype of an instrumented office chair. Also, similar to many behavioral biometrics, permanence of PoPa is not ideal. For example, a user who has a leg, hip or lower-back injury might appreciably alter her posture pattern.

B. Liveness & Replay

In any biometric system used for continuous authentication, liveness detection is a serious concern. For example, a face recognition system needs to detect blinking, breathing, and/or some other artifact of a user being alive and present. Other-wise, as has been demonstrated in the past, it can be subverted by a photo or a mask (face-cast). Traditionally, liveness is attained via some form

of a challenge by the system that requires the user to act. In case of facial recognition, the system might prompt the user to turn her head or look in a particular direction. While this helps achieve liveness and protect against subversion, it also sacrifices transparency and increases user burden.

Fig.11. Assentication setup

Unfortunately, the use of PoPa for de-authentication system also triggers potentially negative implications for user privacy. Unlike traditional single-session authentication techniques, PoPa is inherently tied to the user's physical presence. This has the unintended consequence of leaking whether or not the user is physically at the office or desk. This information could be used by the unscrupulous management to micro-manage and/or audit the time an employee spends at the workstation, the frequency with which they get up, how long they are gone, and other very personal details.

4. AUTHENTICATION

Overview:

To achieve security in any system authentication of the user plays a major role. In our system to achieve authentication, we use user secret password which is only known to user and system. When the user wants to authenticate with the system user first starts the system. Then the system asks for a password from the user. Then the user enters the password which is known to him. The system takes the password and verifies it with the password of the user which is previously stored in the system. If both matches then the system identifies the user as a legitimate and provides service to the user.

Schematic Diagram:

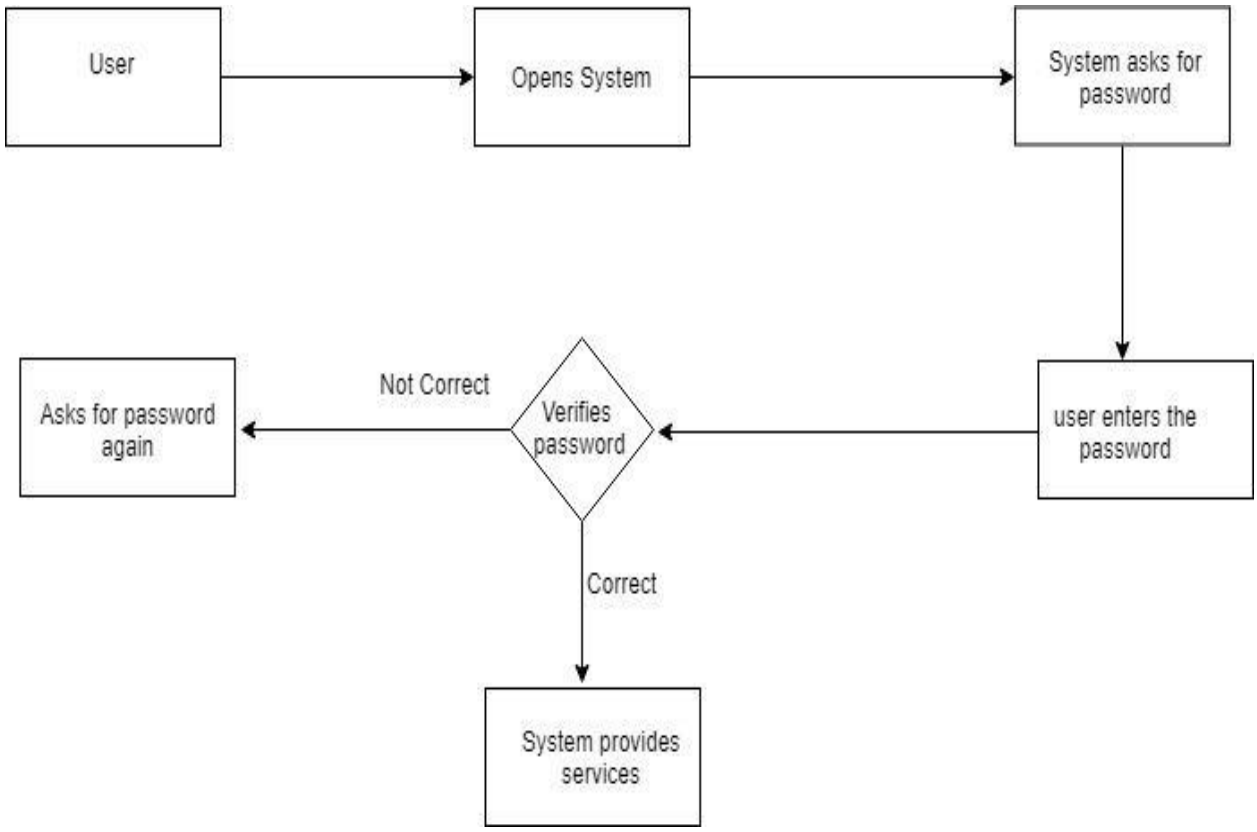

Fig.12. Schematic view of Authentication

As shown in the diagram user first opens the system. Then the system asks for password and user enters it. Then the system verifies the password if the password is correct then the user gets authenticated. If the password is wrong then the system asks to enter the password again.

Implementation:

User starts the program to get services of the system. Then the system asks the user to enter the password to ensure the user as the legitimate one. User enters the password as shown in the below diagrams.

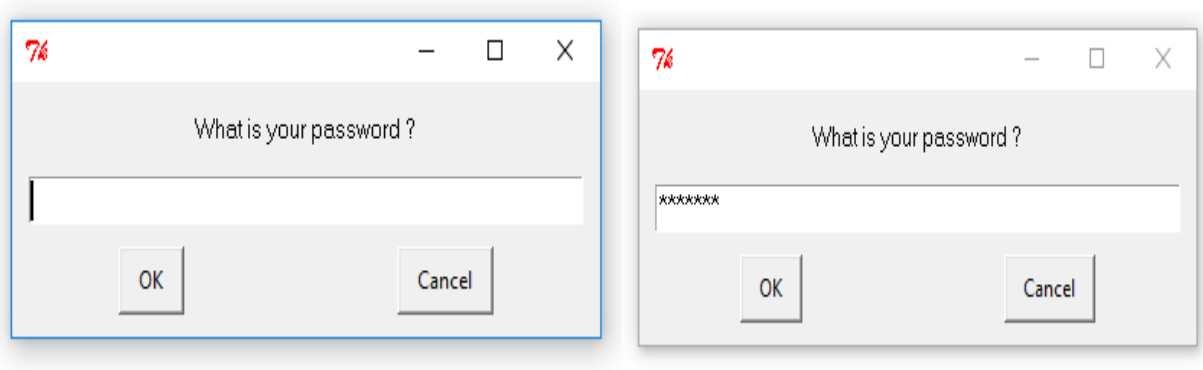

Fig.13.User Authentication

The system takes the input from the user and verifies the password with the user's password which is already stored in the system. The password is only known to the user and system, such that security is ensured. If the password entered by the user is correct then it shows the welcome message as shown in below image.

```
============= RESTART: C:\Users\saisushma\Desktop\project\p8.py =============
welcome!!
```

Fig.14. Successful Authentication

If some intruder is trying to enter the system as the user then he does not know the password. In such cases, they enter the wrong password. Then the system verifies the password and finds it as incorrect and asks the user to re-enter the password. In some cases the legitimate user mistakenly enters the wrong password. In such cases also system asks to re-enter the password again, which is shown in the below diagram.

```
============= RESTART: C:\Users\saisushma\Desktop\project\p8.py =============
your password is incorrect.. pleaese try again!
```

Fig.15. Unsuccessful Authentication

Using a password the system ensures the authentication of the user and provides security to the system.

5. CONTINUOUS MONITORING

Overview

After authentication is done successfully the system takes the temperature values from the sensor through Arduino. System process the temperature values and finds whether the person is present or not. Here the advantage is that the system can capture the moments of the legitimate user. So it helps in continuous monitoring.

Schematic Diagram:

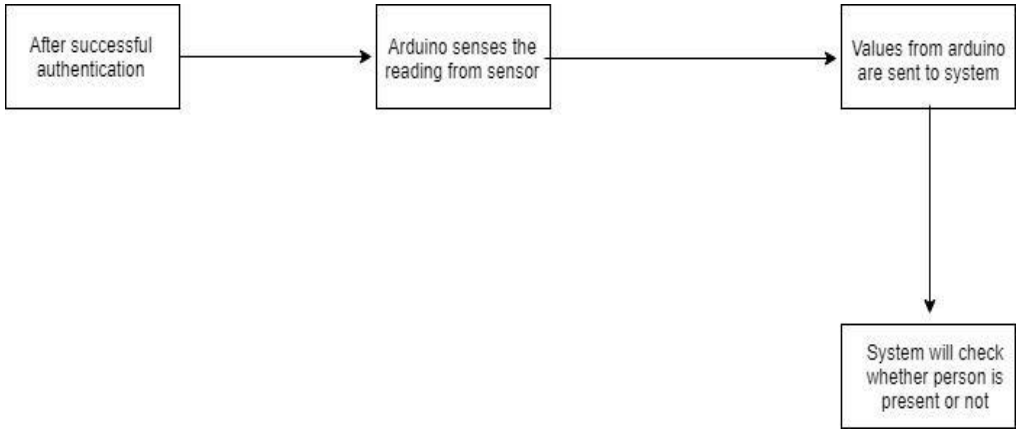

Fig.16. Schematic view of Continuous Monitoring

In our proposed system we achieved continuous monitoring through D6T thermal sensor, which is used to measure the surface temperature of objects. The Silicon lens in Thermal sensor focuses radiant heat (far-infrared rays) emitted from objects onto the thermopile sensor in the module. The thermopile sensor generates electromotive force in accordance with the radiant energy (far-infrared rays) focused on it. The values of this electromotive force and the internal thermal sensor are measured. Then, the device calculates the measured value (temperature of the object) via an interpolation calculation that compares the measured values with an internally stored lookup table. The measured value is output via the I2C bus, and read using a host system.

There are different types in the sensor. In our proposed system we used two sensors D6T-44L-06 and D6T-8L-09. The only difference between the two sensors is field of view. In the D6T-44L-06 sensor we will get 16 temperature values (4*4 grids). Whereas in D6T-8L-09 we get 8 temperature values (8*1 grids).

We used arduino to read the values from the sensor. First the circuit is connected to the arduino using breadboard as per the circuit diagram shown below.

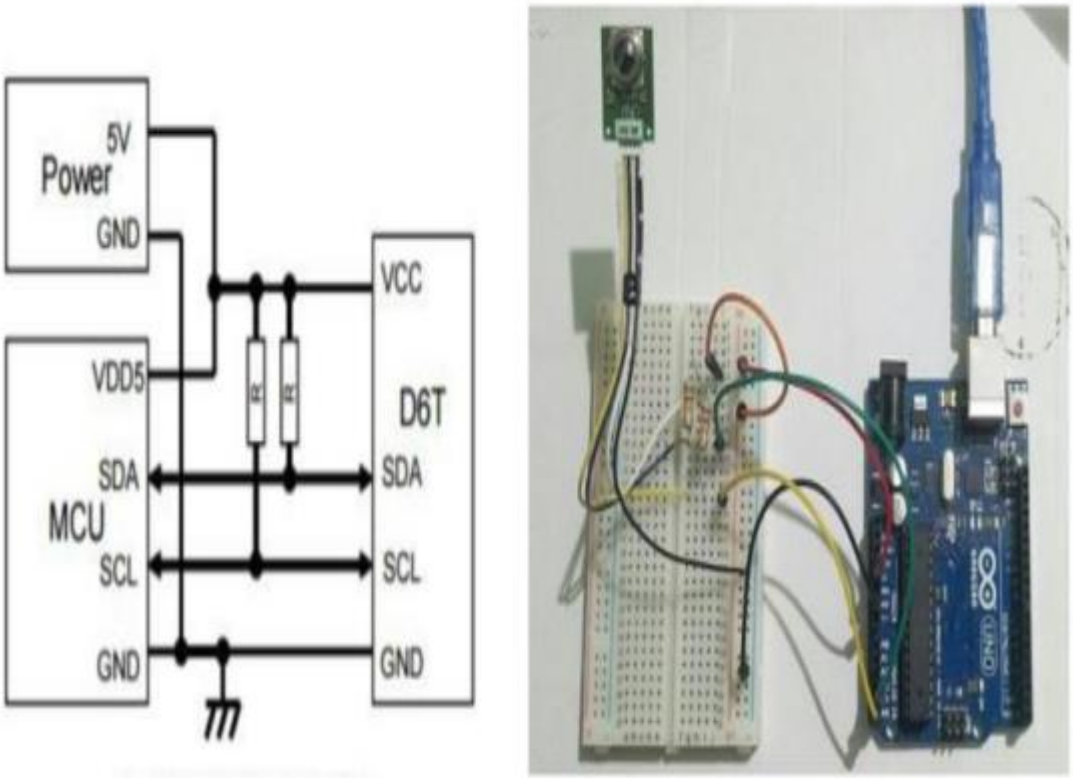

Fig.17. Circuit Diagram

The Arduino is connected to the system. The code which is written in the Arduino is verified and then uploaded to the Arduino board. Through the Arduino board, the temperature values read from the sensor is sent to the system which we can see through the serial monitor in the Arduino. The temperature values observed from the D6T-8L-09 sensor through the serial monitor is shown below.

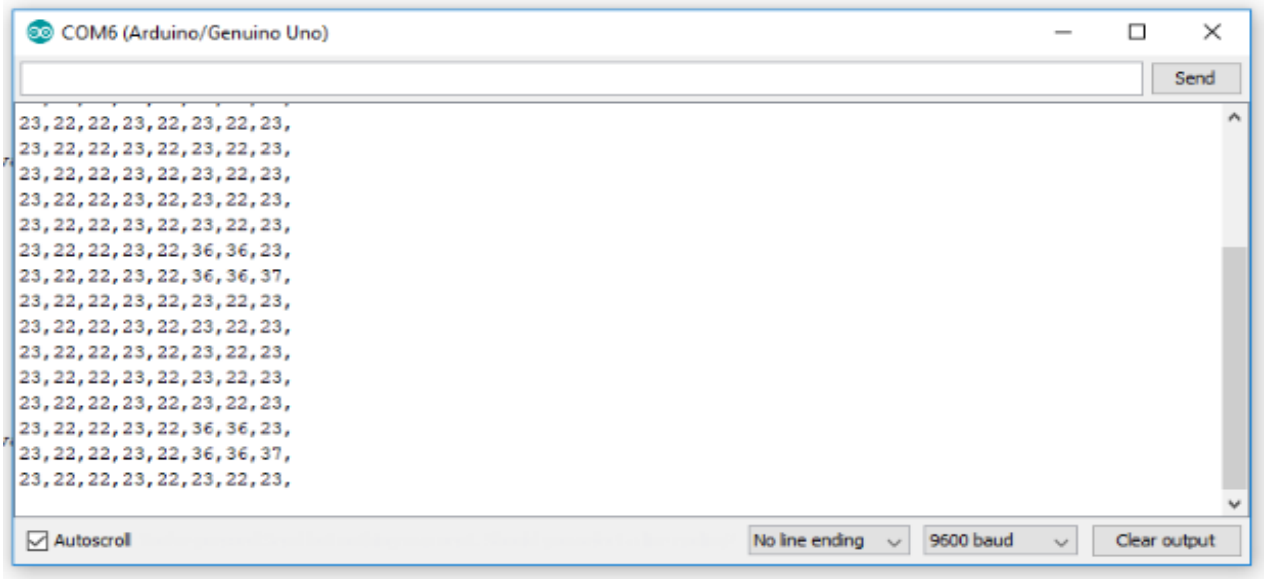

Fig.18. Output in Serial Monitor

The values from the Arduino is taken by the system through the serial port. The data which is read will be in the form of 'utf-8'. The system converts them into integers and finds the presence of the user before the system. The values read by the system from Arduino after authentication is shown in the following diagram.

```
============= RESTART: C:\Users\saisushma\Desktop\project\p8.py =============
welcome!!


23 22 22 23 22 36 36 37
```

Fig.19. Successful transmission of output of D6T using Serial Port

6. DE-AUTHENTICATION

Overview

As discussed earlier, the temperature values from the sensor are read by the system through Arduino. Normal human body temperature, also known as normothermia or euthermia, is the typical temperature range found in humans. The normal human body temperature range is typically stated as 36.5–37.5 ° C. In India, for scientific work, room temperature is taken to be about 20 to 25 degree Celsius with an average of 23° C.

Schematic Diagram:

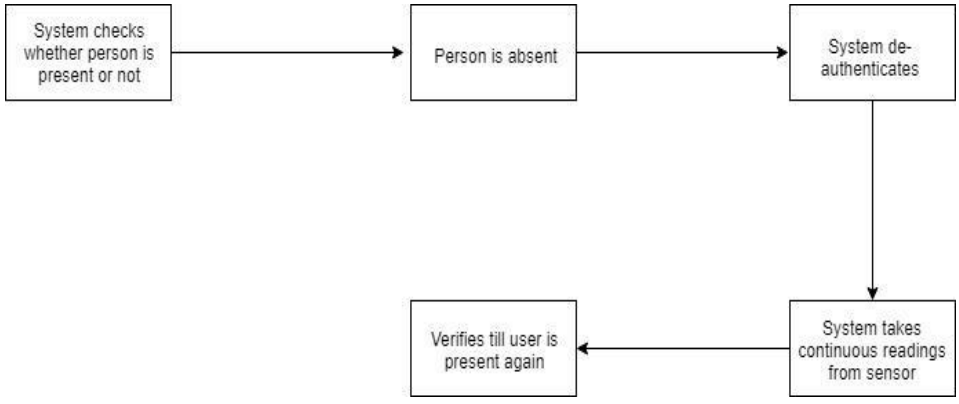

Fig.20. Schematic view of De-authentication

The system is able to decide the presence of human using room temperature and human body temperature.

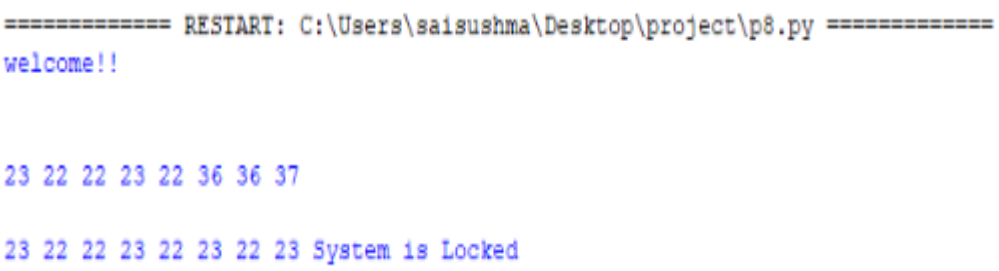

Fig.21. Output of De-authentication

In the above diagram after authentication is done successfully, the system starts taking temperature values from Arduino. As we use D6T-8L-09 we got only 8 values. Each value of temperature at each reading is the temperature of each grid of the field of view. In the first reading, it has 36, 36, 37 reading which means that human is present in the field of view. In the second reading from the Arduino, all 8 temperatures belong to room temperature. It shows the absence of the legitimate user before the system. Then the system gets locked as shown in the image. It indicates that re-authentication is done successfully in the absence of the legitimate user.

Even after de-authentication, the system reads the temperature values of the sensor through Arduino, for re-authentication. While reading the temperature values from the sensor if the system founds the human it prepares the system for re-authentication. Reading values after de-authentication is shown in the following diagram.

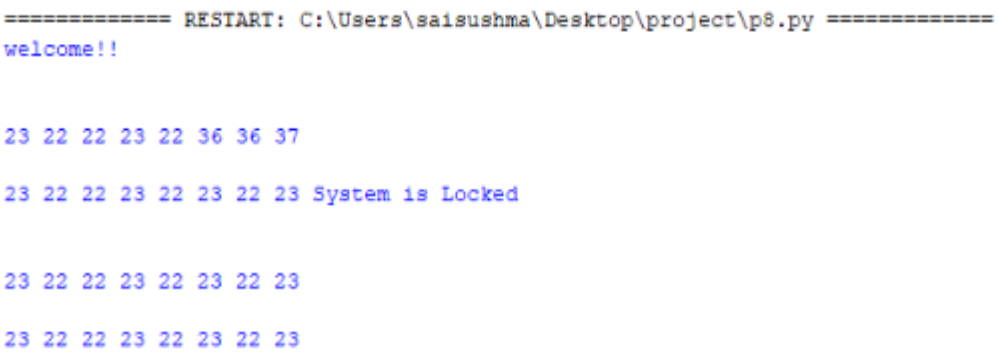

Fig.22. Continuous monitoring after De-authentication

7. RE-AUTHENTICATION

Overview:

After de-authentication is done the system still takes the input from the sensor to find the presence the human again. If he founds the presence of human through temperature values, then the system starts building the model for face recognition. After building the model the system captures the image through webcam. If the image matches with the user's image which is stored previously in the system proceed further for speech recognition. A captcha is shown, the user has to spell the letters as the hidden password. If hidden password matches with the text user spell the system provides the service. If any step fails the user is asked for login again.

Schematic Diagram:

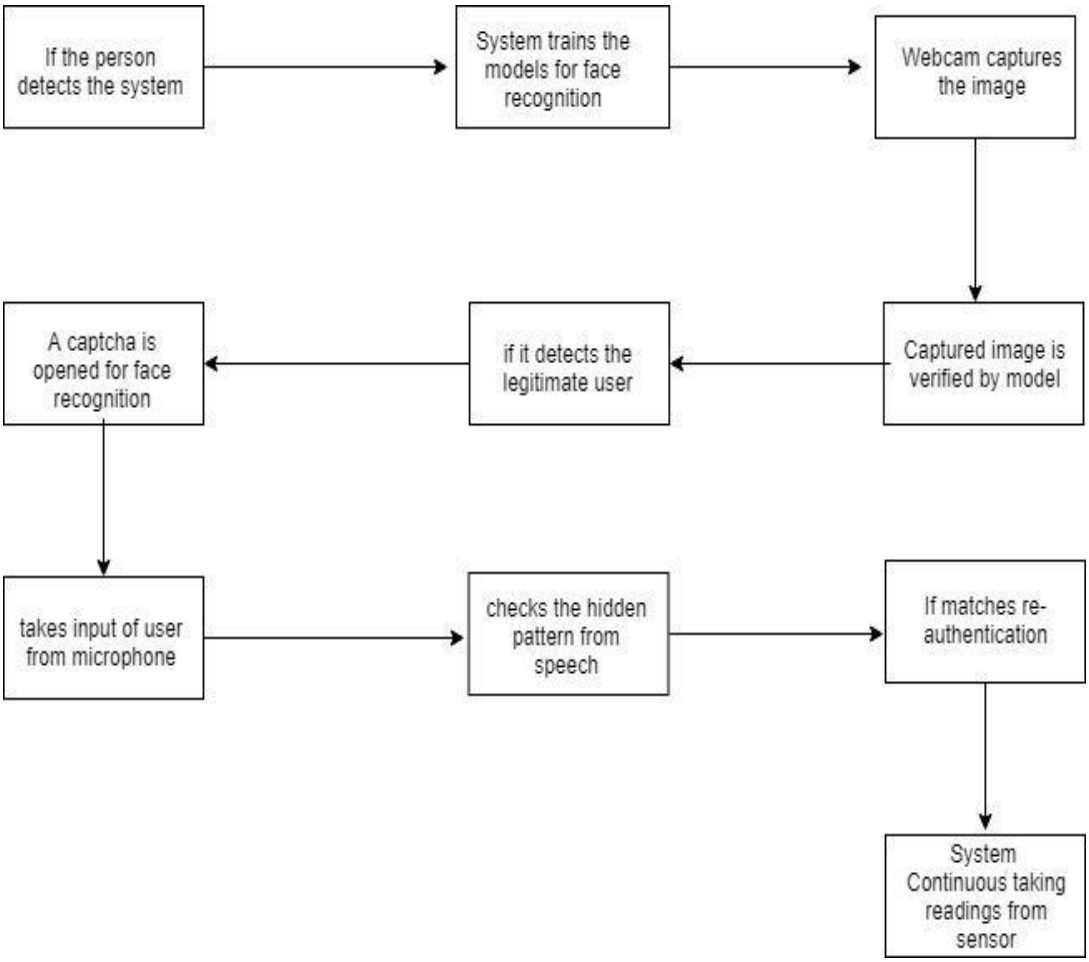

Fig.23. Schematic view of Re-authentication

As shown in the diagram the system takes the readings from the arduino, and finds the presence of human. If found then system starts for building the model as shown in the following picture.

```
============= RESTART: C:\Users\saisushma\Desktop\project\p8.py =============
welcome!!


23 22 22 23 22 36 36 37

23 22 22 23 22 23 22 23 System is Locked


23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 36 36 23

Found person..Syetem is getting ready for Reauntecation
Preparing data...
```

Fig.24. Building model for re-authentication

In the above picture in last reading the temperature values 36, 36 tells the presence of the human. Then the system starts for building the model. After building the model the webcam tries to capture the image. If the face is not recognized in front of the webcam then webcam doesn't capture the image. When it finds the face in webcam then the only system captures the image. After capturing the image system predicts the image using the model previously build. There will be two cases in this case, one is observed user is the legitimate user and the other case is that observed face is an intruder's face.

In our experiment, ‘Shushma’ is the legitimate user where ‘XYZ’ isn’t. If the observed face is the image of Shushma then system further proceeds for speech recognition, which is shown in the following diagram.

```
============= RESTART: C:\Users\saisushma\Desktop\project\p8.py =============
welcome!!


23 22 22 23 22 36 36 37

23 22 22 23 22 23 22 23 System is Locked


23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 36 36 23

Found person..Syetem is getting ready for Reauntecation
Preparing data...
Data prepared
('Total faces: ', 48)
('Total labels: ', 48)
Predicting images...
Prediction complete
SHUSHMA
```

Fig.25. Successful face recognition

If the observed face is not the face of shushma, then the system feels some unauthorized person is trying to access the system and terminates the system and asks the user to login again. Following diagram shows that the observed face is an image of some 'XYZ' and asks to login again.

```
============ RESTART: C:\Users\saisushma\Desktop\project\p8.py ============
welcome!!


23 22 22 23 22 36 36 37

23 22 22 23 22 23 22 23 System is Locked


23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 36 36 23

Found person..Syetem is getting ready for Reauntecation
Preparing data...
Data prepared
('Total faces: ', 48)
('Total labels: ', 48)
Predicting images...
Prediction complete
XYZ
system doesnt recognised you ..Login again!!
```

Fig.26. Unsuccessful face recognition

Though face recognition is done successfully for re-authentication, it is not sufficient for a high-security system. Because face recognition is prune to many attacks and imitations. To increase the security in the system we used two-factor authentication where the second factor is speech. To achieve this, after successful recognition of the legitimate user by the face recognition system shows a captcha image. The sample captcha is shown below

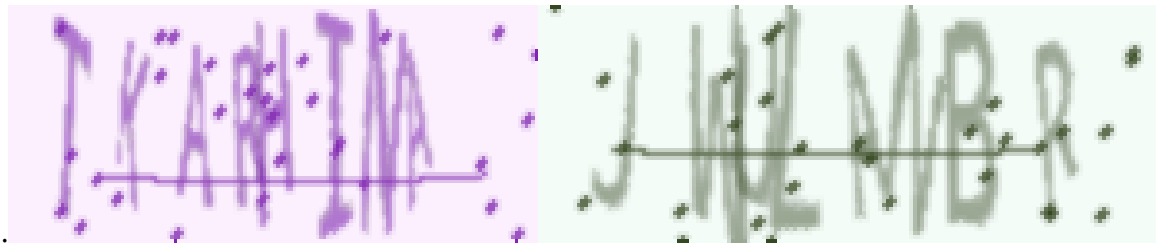

Fig.27. Sample Captcha

In our system, we build captcha only using capital letters. To increase the complexity we can also use small letters and numerals. Captcha is used to prevent bots. But captcha cannot control human attacks. For example, other persons (not legitimate user) also can read the captcha. To prevent this we used a hidden pattern while verifying the captcha spell by the user. In our system, the legitimate user has to spell only last letter, first letter and third letter from the last respectively.

This hidden pattern is only known to the legitimate user. So this ensures security from the human attacks.

To our simplicity the captcha is printed in the terminal in the following images.

```
============= RESTART: C:\Users\saisushma\Desktop\project\p8.py =============
welcome!!


23 22 22 23 22 36 36 37

23 22 22 23 22 23 22 23 System is Locked


23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 36 36 23

Found person..Syetem is getting ready for Reauntecation
Preparing data...
Data prepared
('Total faces: ', 48)
('Total labels: ', 48)
Predicting images...
Prediction complete
SHUSHMA
X
U
D
D
O
Z
L
B
listening
System Predicts:b a z
Not Matched!!
you are failed to say the password ,,please Login Again
```

Fig.28. Unsuccessful speech recognition

In the above case according to the hidden pattern user have to spell ‘B X Z’ respectively. But the user spelt wrongly. As a result the hidden pattern does not matched and system asked to login again. Here for simplicity we considered case in-sensitive while checking the patterns.

The following image shows the case that user spelt the hidden pattern correctly. In such cases re-authentication is done successfully. And system continues its services to the user.

```
============= RESTART: C:\Users\saisushma\Desktop\project\p8.py =============
welcome!!


23 22 22 23 22 36 36 37

23 22 22 23 22 23 22 23 System is Locked


23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 23 22 23

23 22 22 23 22 36 36 23

Found person..Syetem is getting ready for Reauntecation
Preparing data...
Data prepared
('Total faces: ', 48)
('Total labels: ', 48)
Predicting images...
Prediction complete
SHUSHMA
N
U
B
J
D
I
Q
V
listening
System Predicts:VN I
matched!!
System Unlocked
```

Fig.29. Successful speech recognition

While comparing the hidden patterns with the user spelled letters the system considers case insensitive and does not even consider the spaces between letters. Thus re-authentication is done successfully using speech and face recognition.

8. RESULTS AND DISCUSSION

A secured system which provides authentication, de-authentication and re-authentication for continuous monitoring is developed. Authentication is provided by using the password of the user which is considered as highly confidential as it is known only to the user and system. Thus authentication is achieved successfully. After successful authentication of the user with the system, continuous monitoring is achieved by using the D6T thermal sensor which reads the temperature values of its field of view continuously and gives to the system using serial port. This ensures continuous monitoring of the user. De-authentication is done when the values read by the sensor does not lie in the range of human body temperature. This leads to successful de-authentication when the legitimate user is not using the system. Re-authentication starts when the human comes into the field of view of the sensor. Re-authentication is done in a more secure manner using facial recognition and speech recognition. To develop a system free from imitation attacks hidden patterns are included in the speech recognition. Hence a system which provides continuous monitoring using authentication, de-authentication and re-authentication is developed.

9. CONCLUSIONS AND FUTURE SCOPE

Today most of the systems have advanced authentication mechanisms to verify the user at session start, which results in security breaches for the remainder of the session. We proposed an architecture which continuously monitors the user. We achieved this using authentication, de-authentication, and re-authentication mechanisms. The proposed system is able to provide services only to the legitimate user throughout the session. Our proposed architecture works well with the personal system, but the idea can be extended into various applications such as Gmail login, social media accounts to improve the security and privacy of the user.

REFERENCES

[1] Kaczmarek, Tyler, Ercan Ozturk, and Gene Tsudik. "Assentication: User De-authentication and Lunchtime Attack Mitigation with Seated Posture Biometric." In *International Conference on Applied Cryptography and Network Security*, pp. 616-633. Springer, Cham, 2018.

[2] Anusas-amornkul, Tanapat, and Kasem Wangsuk. "A comparison of keystroke dynamics techniques for user authentication." In *2015 International Computer Science and Engineering Conference (ICSEC)*, pp. 1-5. IEEE, 2015.

[3] Dinca, Lavinia Mihaela, and Gerhard Petrus Hancke. "The fall of one, the rise of many: a survey on multi-biometric fusion methods." *IEEE Access* 5 (2017): 6247-6289.

[4] Srivastava, Stuti, and Prem Sewak Sudhish. "Continuous multi-biometric user authentication fusion of face recognition and keystoke dynamics." *Humanitarian Technology Conference (R10-HTC), 2016 IEEE Region 10*. IEEE, 2016.

[5] Liu, Caixia. "The development trend of evaluating face-recognition technology." *Mechatronics and Control (ICMC), 2014 International Conference on*. IEEE, 2014.

[6] Barbu, Tudor, Adrian Ciobanu, and Mihaela Luca. "Multimodal biometric authentication based on voice, face and iris." *E-Health and Bioengineering Conference (EHB), 2015*. IEEE, 2015.

[7] Eberz, S., Rasmussen, K.B., Lenders, V., Martinovic, I.: Preventing lunchtime attacks: fighting insider threats with eye movement biometrics. In: NDSS (2015).

[8] Alves, Diego D., G. Cruz, and C. Vinhal. "Authentication system using behavioral biometrics through keystroke dynamics." *Computational Intelligence in Biometrics and Identity Management (CIBIM), 2014 IEEE Symposium on*. IEEE, 2014.

[9] Chinchu, S., Anisha Mohammed, and B. S. Mahesh. "A novel method for real time face spoof recognition for single and multiple user authentication." *Intelligent Computing, Instrumentation and Control Technologies (ICICICT), 2017 International Conference on*. IEEE, 2017.

[10] Obin, Nicolas, and Axel Roebel. "Similarity search of acted voices for automatic voice casting." *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24.9 (2016): 1642-1651

[11] Bours, Patrick, and Soumik Mondal. "Continuous authentication with keystroke dynamics." *Norwegian Information Security Laboratory NISlab* (2015): 41-58.