Security by user through application to lock/unlock machine by face detection

P.S.Hanawate[1], Aishwarya Patil[2], Saloni Kulkarni[3] Kalyani Kolte [4]

[1] computer department
NBNSSOE
pune
poonamkumar.hanwate@sinhgad.edu
[2] computer department
NBNSSOE
pune
aishwarya.patil1630@gmail.com
[3] Computer department
NBNSSOE
pune
saloni.kulkarni21@gmail.com
[4] Computer department
NBNSSOE
pune
kalyanikolte9196@gmail.com

---***---

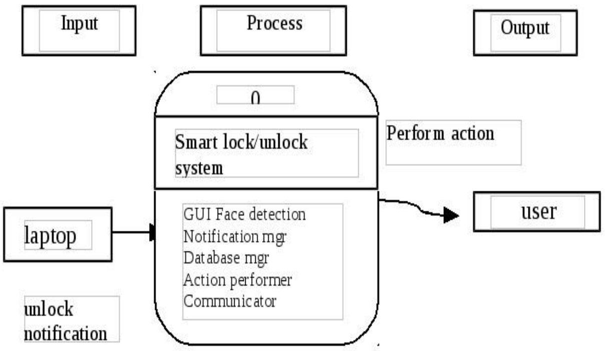
Abstract - *The system will be designed for security purposes. It will work if anybody tries to unlock the laptop, it will act as a trigger to the camera and the camera will capture the image of the person standing in front of the Laptop. Then application running on laptop will send notification to android app about this event. This will help user to decide what action he/she should take. The system is designed such that the motion of the user will be captured from the camera and the user will be detected and then only he will be given a key to lock or unlock. The application was designed to allow the user to also check the status of the door. The mobile device requires a password to increase the security of the system. Security is main concern for handling such documents in laptop mobile devices computers. We do not have that much good security with these devices. The security we have is making lock for system as a password but anyone can crack the password and harm our data to avoid this situation so many security functions are provided like thumb print, retina recognition etc. But there is no security for checking that who is trying to crack the password. That security is provided in our proposed system.*

1.INTRODUCTION

Security is main concern for handling such documents in laptop mobile devices computers. We do not have that much good security with these devices. The security we have is making lock for system as a password but anyone can crack the password and harm our data to avoid this situation so many security functions are provided like thumb print, retina recognition etc. But there is no security for checking that who is trying to crack the password. That security is provided in our proposed system.

1.1 System overview

A. system architecture

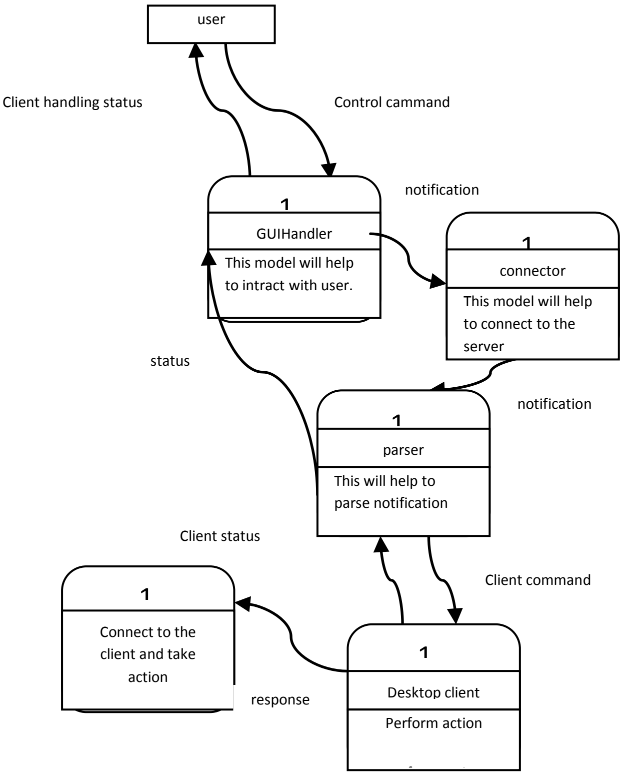


B. Architecture flow

1. *Input:*
   - password entered by client machine
   - photograph of machine user
   - actions perform by machine user
   -

2. *process:*
   - lock/unlock machine
   - face detection
   - authentication
   - actions handling
3. *output:*
   - authentication
   - action execution

C. Architecture description



Machine user wants to login. He/she enter the password. Then request will be send to GUI handler. GUI is (graphical user interface) handler which helps to interact with user. GUI handler will notify to connector. Connector helps to connect the server. Then connector notify to parser which will help to parse the notification. Then it will command to desktop client. Client performs action then as response action is taken. Then at the same time client status is send to parser. After that parser send status to GUI handler and then GUI handler to user.

D. Scope of Project

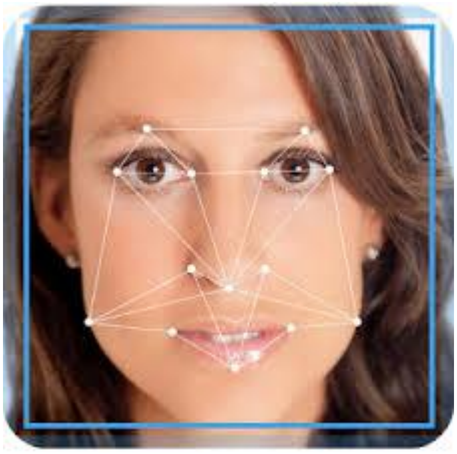
1. The system will be designed for security purposes.
2. When we leave your laptop, all we have to do is lock your system and our application starts the protection.
   - If anybody unlock our system, our application will takes his/her images via laptops camera
   - Send notification to laptop owner (via Wi-Fi/Internet) along with images
   - Laptop Owner will receive this as notification on his/her Smartphone
   - Laptop Owner can take following actions:
3. Lock computer again remotely
4. Retrieve critical data from the Laptop
5. Immediately log user out, change admin privileges and password.
6. Delete or encrypt sensitive files or directories
7. Capture images from built in camera
8. Real-time Desktop Capture
9. Take screen shots .

E. Algorithms

1. Eigen vector(for face detection)
2. REST (for communication)
3. Flow of system

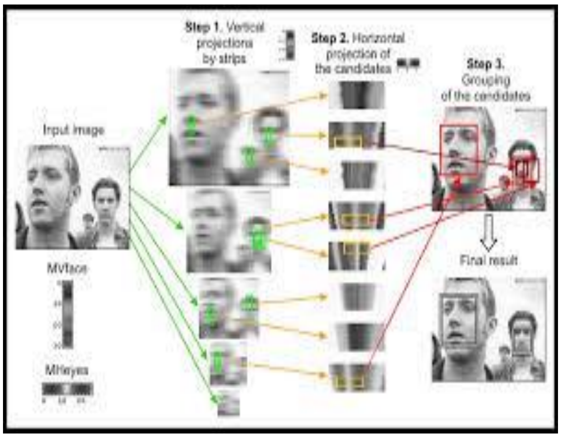
2. Face Detection

Face detection is a technique used to recognize human faces in digital form of images. This is one of the computer technology used in variety of applications in real world. In this paper we are using this technique to know who the person is getting the machine password. For that we are using machine camera. Through that camera we have to capture image of person who is in front of machine. To capture the image of that person we need face detection. The person can be located in any sense so we have to detect and capture the image. There are many categories of face detection. It depends upon eye, nose, mouth etc position or gender classification. It can also be based on color or image. segmentation. Following figure shows the points detected on face and then image get captured.



- It is computer technology which is used in various applications for the identification of human faces in digital images.
- There are two techniques of genetic algorithm and eigenfaces for the face detection.
- Each possible face candidates is normalized to reduce lightning effect caused due to uneven illumination and the shirring effect due to head movement. The fitness value of each candidate is measured based on its projection on the eigen-faces. After a number of iterations, all the face candidates with a high fitness value are selected for further verification. At this stage, the face symmetry is measured and the existence of the different facial features is verified for each face candidate.
- for the analysis purpose we need to assume the centered face image and same size as training face and eigenfaces.
- by using motion detecting and head tracking face can be detect easily.
- people are constantly moving, like nods our head and adjust our body hence for the detection of face we can use simple motion detection and tracking algorithms.
- After threshold the filtered image for production of binary motion image, we need to analyze the "motion blobs" that means the person is moving and to determine the head position". for e.g. on the larger blob head is smaller blob and head motion must be reasonably slow and contiguous.
- On the basis of number of images from motion it helps to detect the face easily.

Following figure shows process of face detection. While we have input image the step one in process is vertical projection by strips it detects the face by its several parts. Then after that step two is horizontal projection of the candidates then both the results will be combined in step three. Then final result will be generated. In following image two faces will be detected as a result.


Step 1. Vertical projections by strips
Step 2. Horizontal projection of the candidates
Step 3. Grouping of the candidates
Input image
MVface
MHeyes
Final result

1. Eigen vector

• Main idea behind eigenfaces

- Suppose □□is an *N*2x1 vector, corresponding to an *N*x*N* face image *I* .

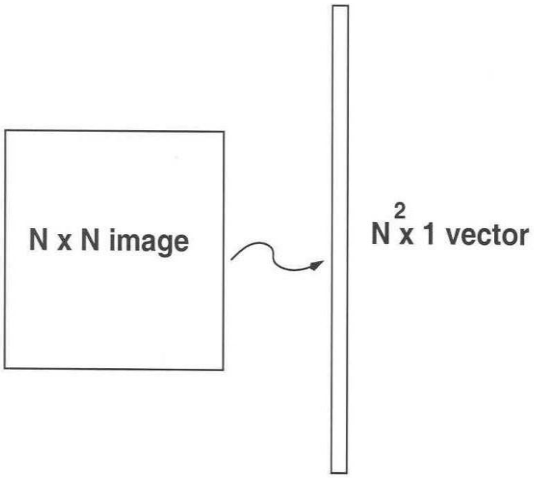
- This idea represent □□(□□□□□ mean face) into a low-dimensional space.

-Represented in equation form as given bellow

□□□□□*mean* □□*w*1*u*1 □□*w*2*u*2 □□. . . *wK uK* (*K*<<*N*2)

- Calculating eigenfaces

Let face image I(x, y) 2D N by N array of 8-bit intensity values. Image is vector dimension N^2 means 65535 dimensional space ensemble of image the map collection point in this space for example figure shows training set of various images of faces. Principle component analysis used to find vectors. These vectors find face space with length N^2.

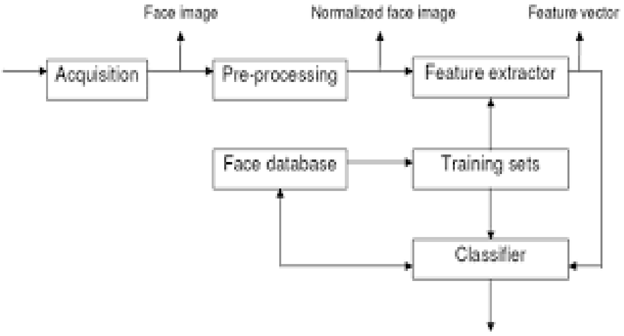


Database

-2500 face images digitized under monitor and controlled conditions.

-16 subject with three head orientation are digitized and also with head size, scales, three lightning conditions etc.

-Gaussian pyramid of 6 level having image resolution 512x512 pixel to 16x16 pixel.



This diagram shows database position in face detection process.

Summary of steps for eigenfaces: Collect a set of characteristics which contain set of images

$$\lambda_k = \frac{1}{M}\sum_{n=1}^{M}\left(\mathbf{u}_k^T\Phi_n\right)^2 \qquad (1)$$

$$\mathbf{u}_l^T\mathbf{u}_k = \delta_{lk} = \begin{cases} 1, & \text{if } l = k \\ 0, & \text{otherwise} \end{cases} \qquad (2)$$

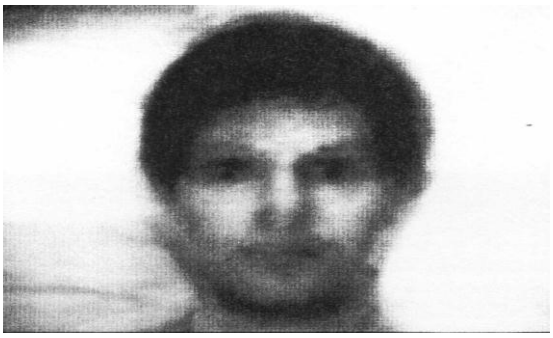
λ_k and u_k are eigenvectors and eigenvalues respectively which we have calculated in equation 1 & 2.

$$C = \frac{1}{M}\sum_{n=1}^{M}\Phi_n\Phi_n^T \qquad (3)$$

$$= AA^T$$

Equation 3 gives covariance matrix. Which calculate 40x40 matrix and find eigenvector and eigenvalues

Which ultimately gives average face Ψ.



From input images shown seven eigenfaces calculated and from those images one average image is calculated. Combine set of normalized training set call as uk. following equations represents this procedure.

$$A^TA\mathbf{v}_i = \mu_i\mathbf{v}_i \qquad (4)$$

$$AA^TA\mathbf{v}_i = \mu_iA\mathbf{v}_i \qquad (5)$$

$$\mathbf{u}_l = \sum_{k=1}^{M} \mathbf{v}_{lk}\Phi_k, \qquad l = 1,\ldots,M \qquad (6)$$

This image gives variation in head size, three light conditions, and three head orientation.

Following equation shows how to classify a face image by eigenfaces.

$$\omega_k = \mathbf{u}_k^T(\Gamma - \Psi) \qquad (7)$$

For each u_k calculate class vector Ω_k by taking average of Ω choose threshold maximum allowable distance $\Theta_€$. For each new face image calculate pattern vector Ω $€_k<\Theta_€$ is minimum distance. $€<\Theta_€$ implies individual is from class vector Ω_k $€_k>\Theta_€$ but distance $€_k<\Theta_€$ then image may be declared as unknown.

$$\epsilon_k^2 = \|(\Omega - \Omega_k)\|^2 \qquad (8)$$

$$\epsilon^2 = \|\Phi - \Phi_f\|^2 \qquad (9)$$

If image is known then it will get added with original set of known faces.

Computation of the eigenfaces

Step 1: obtain face images *I*1, *I*2, ..., *IM* (training faces)

(very important: the face images must be *centered* and of the same *size*)

Step 2: represent every image *Ii* as a vector �*i*

Step 3: compute the average face vector���

���1�*MMi*�1���*i*

Step 4: subtract the mean face:

�*i* ���*i* ��

Step 5: compute the covariance matrix *C*:

C �1*MMn*�1���*n*�*Tn*��*AAT* (*N*2x*N*2 matrix)

Step 6: compute the eigenvectors *ui* of AA^T

The matrix AA^T is very large --> not practical.

Step 6.1: consider the matrix $AT\,A$ (MxM matrix)

Step 6.2: compute the eigenvectors vi of A^TA

$A^TA\ vi$ ��mi vi

What is the relationship between usi and vi?

$A^TA\ vi$ ��mi vi => $AA^T\,Avi$ ��m

$i\ Avi$ =>$CAvi$���mi Avi or Cui ��m

$i\ ui$ where ui ��Avi

Thus, AA^T and A^TA have the same eigenvalues and their eigenvectorsare related as follows: ui ��Avi !!

Note 1: AA^T can have up to $N2$ eigenvalues and eigenvectors.

Note 2: $A^T\,A$ can have up to M eigenvalues and eigenvectors.

Note 3: The M eigenvalues of $A^T\,A$ (along with their corresponding

eigenvectors) correspond to the M *largest* eigenvalues of AA^T (along

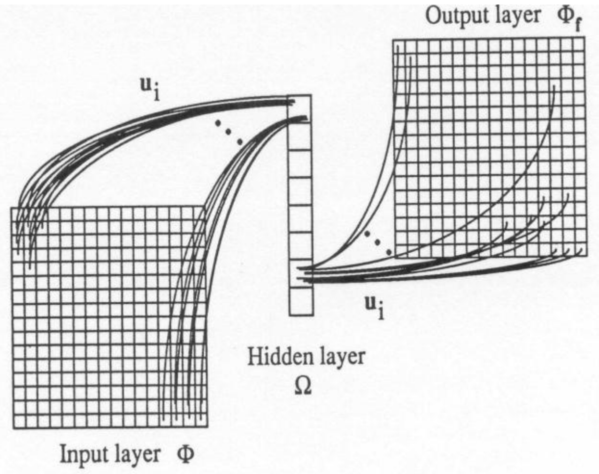
with their corresponding eigenvectors).

Step 6.3: compute the M best eigenvectors of AA^T : ui ��Avi

(important: normalize ui such that ||ui|| ��1)

Step 7: keep only K eigenvectors (corresponding to the K largest eigenvalues)

Three layer linear network for eigenface calculation. The symmetric weights u_i are the eigen faces and the hidden units reveal the projection of the input image Φ onto the eigenfaces the output Φ_f is the face space projection of input image.



2. REST (for communication)

REST stands for Representational State Transfer. (It is sometimes spelled "ReST".) It relies on a stateless, client-server, cacheable communications protocol -- and in virtually all cases, the HTTP protocol is used.

REST is an architecture style for designing networked applications. The idea is that, rather than using complex mechanisms such as CORBA, RPC or SOAP to connect between machines, simple HTTP is used to make calls between machines.

In many ways, the World Wide Web itself, based on HTTP, can be viewed as a REST-based architecture.

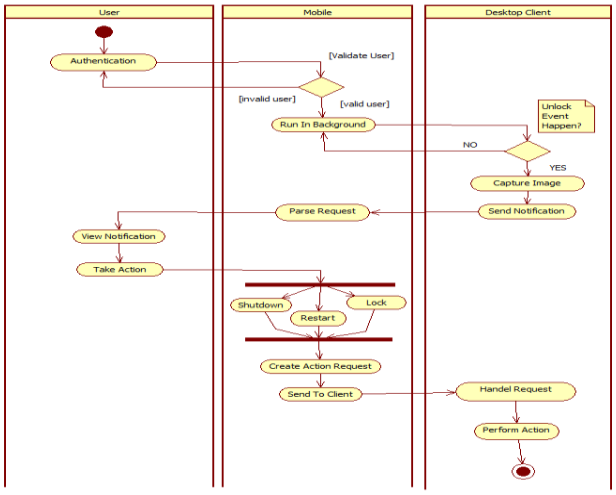
REST full applications use HTTP requests to post data (create and/or update), read data (e.g., make queries), and delete data. Thus, REST uses HTTP for all four CRUD (Create/Read/Update/Delete) operations.

REST is a lightweight alternative to mechanisms like RPC (Remote Procedure Calls) and Web Services (SOAP, WSDL, et al.). Later, we will see how much more simple REST is.

Despite being simple, REST is fully-featured; there's basically nothing you can do in Web Services that can't be done with a RESTful architecture.

- REST is not a "standard". There will never be a W3C recommendataion for REST, for example. And while there are REST programming frameworks, working with REST is so simple that you can often "roll your own" with standard library features in languages like Perl, Java, or C#.

.3. Flow of system:



If user is trusted then user will be authenticated otherwise authentication failure occurs. If unlock event happen then camera act as trigger and capture the image of the person which is standing in front of camera. Then application gets started at background of the machine. It will send the captured image and notification to the user application. Then user is able to take action from given options like lock, restart or shutdown machine. Then action request will be created and send to client. Now request handled by client is executed and action gets performed.

Face Detection Using Eigenfaces

- Given an unknown image G

Step 1: computeF = G -Y

Step 2: compute Fˆ = K

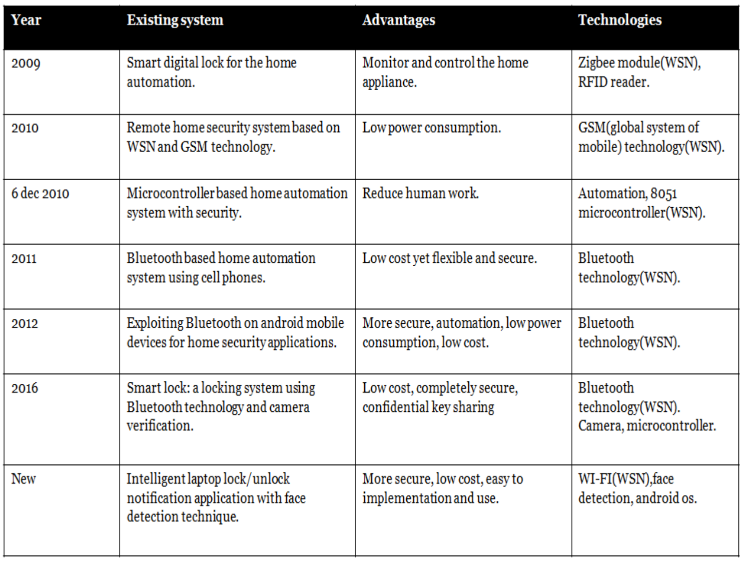
i=1

Swiui (wi = uTiF)

Step 3: compute ed = ||F- ˆF||

Step 4: if ed < Td, then G is a face.

4. Literature survey



Year	Existing system	Advantages	Technologies
2009	Smart digital lock for the home automation.	Monitor and control the home appliance.	Zigbee module(WSN), RFID reader.
2010	Remote home security system based on WSN and GSM technology.	Low power consumption.	GSM(global system of mobile) technology(WSN).
6 dec 2010	Microcontroller based home automation system with security.	Reduce human work.	Automation, 8051 microcontroller(WSN).
2011	Bluetooth based home automation system using cell phones.	Low cost yet flexible and secure.	Bluetooth technology(WSN).
2012	Exploiting Bluetooth on android mobile devices for home security applications.	More secure, automation, low power consumption, low cost.	Bluetooth technology(WSN).
2016	Smart lock: a locking system using Bluetooth technology and camera verification.	Low cost, completely secure, confidential key sharing	Bluetooth technology(WSN). Camera, microcontroller.
New	Intelligent laptop lock/unlock notification application with face detection technique.	More secure, low cost, easy to implementation and use.	WI-FI(WSN),face detection, android os.

5. conclusion

Our proposed system ensures all the quality with respect to security. The system parameters are check with feasibility. This system is able to provide complete security for device. With advance technology of face detection to lock/unlock machine. So that user access security of that machine.

References

[1] [1] http://www.ijtra.com/view/smart-lock-a-locking-system- using- bluetooth -technology-camera-verification.pdf (2016)

[2] [2] Potts, Josh, and SomsakSukittanon. ”Exploiting Bluetooth on Android mobile devices for home security application.” Southeastcon, 2012 Proceedings of IEEE. IEEE, 2012.

[3] [3] Piyare, R., and M. Tazil. ”Bluetooth based home automation system using cell phone.” Consumer Electronics (ISCE), 2011 IEEE 15th International Symposium on. IEEE, 2011.

[4] [4] Kaur, Inderpreet. ”Microcontroller based home automation system with security.” International journal of advanced computer science and applications 1.6 (2010)

[5] [5] M. Turk and A. Pentland, "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, vol.3, no. 1, pp.

[6] 71-86, 1991, hard copy

[7] [6] http://www.face-rec.org/algorithms/PCA/jcn.pdf