

## International Journal of Computer Science and Mobile Computing



A Monthly Journal of Computer Science and Information Technology

ISSN 2320-088X

IMPACT FACTOR: 6.017

*IJCSMC, Vol. 7, Issue. 2, February 2018, pg.89 – 101*

# Real Time Face Recognition System based on KLT Algorithm and Wiener Filtering

**Anupam Baruah<sup>1</sup>, Prof.(Dr.) Lakshmi Prasad Saikia<sup>2</sup>**

<sup>1</sup>Research Scholar, Dept. of Computer Science & Engineering, Assam Down Town University, India

<sup>2</sup>Professor, Dept. of Computer Science & Engineering, Assam Down Town University, India

<sup>1</sup>[anupambaruah04@gmail.com](mailto:anupambaruah04@gmail.com); <sup>2</sup>[lp\\_saikia@yahoo.co.in](mailto:lp_saikia@yahoo.co.in)

---

*Abstract— Main objective of this paper is to represent an approach for biometric recognition of human faces based on KLT algorithm and Wiener filtering techniques. This approach involves registering a set of human faces to a face database, matching real time faces with already stored faces. Matlab based programs are implemented for this approach. KLT algorithm is used for create face database as well as face recognition purpose. Wiener filtering is implemented to separate the illusion-invariant features from face images.*

*Keywords—KLT algorithm, Wiener filtering, Face detection, Face recognition*

---

## 1. INTRODUCTION

Human face detection and recognition is a major topic for modern day researchers. It is very important in many computer fields like criminal identification, access and security, E-banking, Online shopping site, Net banking etc. Real time face detection and recognition is not a simple problem. Many approaches[1] have been already implemented like template matching, neural network, MRC etc., Different algorithm have been used over the past few years. These algorithms have own some disadvantages. The techniques used in our paper are the most effective among those. The algorithms applied in this paper for face detecting and tracking is KLT algorithm and wiener filtering is applied for calculating matching percentage of different features points for recognition process.

### 1.1 KLT algorithm:

Kanade–Lucas–Tomasi (KLT) is an approach to feature extraction in Computer vision. This approach is faster than traditional techniques for comparing best matches between two images. The algorithm was proposed in 1981. The algorithm is used for detecting various features points which helps for tracking required points in a good standard[2]. The algorithm is based on three main steps[3]

- i. Detect a real time face from webcam to track
- ii. Identify various features of the face
- iii. Finally track the face.

Face is detected using vision.CascadeObjectDetector object to detect face location in real time video frame.Cascade object detection uses viola-jones detection algorithm[4].To track the face continuously over the video frame KLT algorithm is used. After detecting face the algorithm tracks a set of features points from the video frames. Vision.pointTrackerSystem is used for initialize a face tracker to find corresponding points in the current frame. The goal of this algorithm is to align the template T(x) to an input image. Fig.1 shows overall flowchart of the algorithm. At the initial point tracker tries to align template with the image. If the template and the image not coverage the tracker will search for another point to find the alignment.

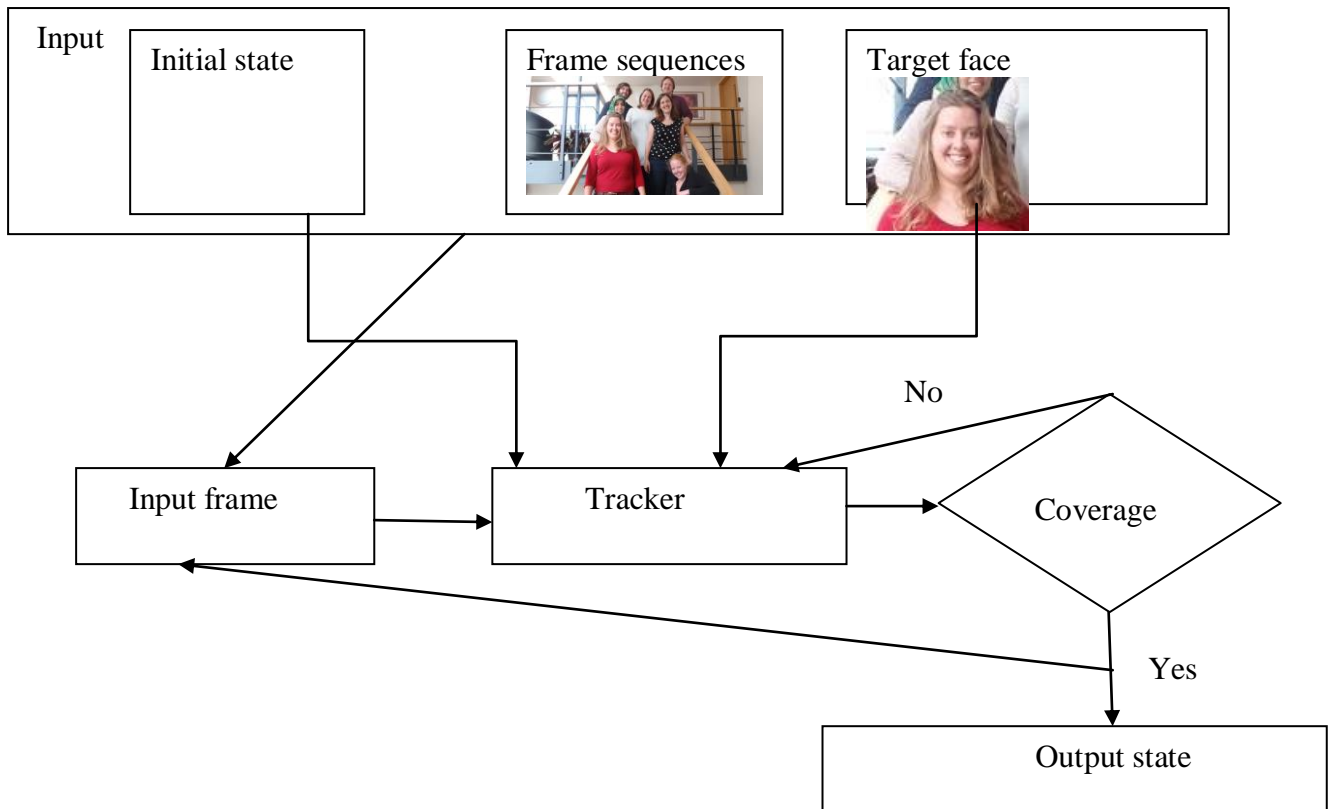


Fig. 1 Diagrammatic representation of the system

This successive process continues until tracker finds the best alignment and a new state will use to initial point for the next frame.template is used for the state estimation in KLT trackers which align a template image T(X) at t=1bto an image I(x) at t=2 , where pixel coordinates are (x;y).Wrap W(x;p) can be calculated as

$$W(x;p) = \begin{pmatrix} ((1+p1).x+p3.y+p5) \\ (p2.x+(1+p4).y+p6) \end{pmatrix} \dots\dots\dots(1)$$

$$= \begin{pmatrix} 1+p1+p3+p5 \\ P2+1+p4+p6 \end{pmatrix} \begin{pmatrix} X \\ Y \\ I \end{pmatrix} \dots\dots\dots(2)$$

$$\text{Where } p=(p_1, \dots, p_n) \dots \dots \dots (3)$$

$$= \sum_x [I(W(x;p)) - T(x)]^2 \dots \dots \dots (4)$$

Equation 4 is carried out on all the pixels  $x$  in the template  $t(x)$  and minimization is gained with respect to  $p$ . In case of non linear, The KLT assumes that the current estimate of  $p$  exists and iteratively searches for the new  $p$  by following expression :

$$= \sum_x [I(w(x;p+\Delta p)) - T(x)]^2 \dots \dots \dots (5)$$

$p$  can be updated by  $p \leftarrow p + \Delta p$

The algorithm finds the best alignment for each possible location in the search space.[5]

**1.2 Wiener filtering:**

Weiner filter was proposed by Norbert Wiener in 1940 and was published by 1949 [6]. This is a statistical approach for estimate frequency domain by transforming the degraded image into the frequency domain using Discrete Fourier Transform (DFT), filtering and inverse transforming[7]. Noise of a corrupted signal can be calculated using wiener filtering. It is the linear estimation of of an original image. It minimizes the overall mean square error in the process of inverse filtering and noise smoothing Frequency domain of wiener filtering is given by

$$H(u,v) = \frac{P_{d(u,v)}}{P_{d(u,v)} + P_{N(u,v)}} \dots \dots \dots (1)$$

Where  $p_{d(u,v)}$  and  $p_{N(u,v)}$  represents the power spectra [8].

**1.3 Face detection:**

It is a concept for detecting faces from individuals or group image. In matlab code face detection can be done by cascade object detector with set of facial features.

**1.4 Face recognition:**

It is a concept for determining the presence of a face in face database. Face recognition is done by calculating percentage of matching features point of both the images.

## 2. RELATED WORKS

In 2015, Ekta Thaman, Ramanpreet Kour proposed an approach for a facial identification and tracking system for security purpose using spatio-temporal segmentation. Matlab technology used which allows immediate identification and strong detection of face in high-resolution videos. [9]

In 2015, Prof. Rushabh Shah ,Aalisha Sheth,Biyanta Shah,Ayushi Shah ,Dr. Priyanka Sharma proposed various types of techniques which are used to detect and track faces. Techniques they have used are KLT algorithm and CAM-Shift. They have used skin tone as a feature and converted the first frame to hue channel data and then detects the face in the rest of the video[10].

In 2015, Kapil, Anurag Singh have analysed two techniques cam shift and KLT algorithm and as per the visible results from the figures 1 and 2 we conclude that KLT algorithm is more accurate and advantageous than cam shift technique.[13]

In 2016, Ritesh Boda and M. Jasmine Pemeena Priyadarsini proposed two approaches for detecting and tracking human faces. They applied viola jones algorithm to detect faces and used KLT algorithm for tracking the face.[11].

In 2017, P.J Leo Evenss, Jennings Mcenroe .S, A.Prabhu Chakkaravarthy proposed an approach for face recognition using Viola jones and KLT algorithm.[12]

In 2015, Raajan.P,Muthuselvi.S,Agnes Saleema.A proposed an approach for preprocessing of images and compressing of filtered images with lossy and lossless compression is presented.Preprocessing is carried out with weiner filtering and compression of images with Harr segmentation[14]

In 2015, Prodip Biswas,Abu Sufian sarkar,Muhammed Mynuddin proposed an image processing method in Matlab environment. Then proved how an image can be effected by noise and how that noise is removed from image.[15]

## 3. METHODOLOGY

The overall system consists of two processes. One is training process and other is testing process. Training process is consists of following steps:

- a) Real time face capturing using webcam
- b) Detect face
- c) Extract good features
- d) Track features
- e) Track faces
- f) Store tracked faces to face database

The testing process consists of following steps:

- i. Real time face capturing using webcam
- ii. Detect face
- iii. Extract good features
- iv. Track features
- v. Compare extracted features with features extracted from face database

### **3.1 Training process:**

#### **3.1.1 Real time Image capturing:**

Image capturing is done by the use of webcam where user have to stand in front of camera .captured images are send to Matlab for detecting and tracking image .

#### **3.1.2 Detect faces:**

Faces which are captured from webcam are detected using viola jones algorithm using cascade object detector class. Here faces are cropped along with thier bounding boxes and pass to features extraction process.

Following are the Matlab code for detection of faces using bounding box:

```
% Create a cascade detector object.
faceDetector = vision.CascadeObjectDetector();

% Read a video frame and run the face detector.
videoFileReader = vision.VideoFileReader('tilted_face.avi');
videoFrame      = step(videoFileReader);
bbox            = step(faceDetector, videoFrame);

% Draw the returned bounding box around the detected face.
videoFrame = insertShape(videoFrame, 'Rectangle', bbox);
figure; imshow(videoFrame); title('Detected face');

% Convert the first box into a list of 4 points
% This is needed to be able to visualize the rotation of the object.
bboxPoints = bbox2points(bbox(1, :));
```

#### **3.1.3 Extract good features:**

After detecting faces KLT algorithm tracks a set of feature point across the video frames. Following are the matlab code for extract good features.

```
% Detect feature points in the face region.
points = detectMinEigenFeatures(rgb2gray(videoFrame), 'ROI', bbox);

% Display the detected points.
figure, imshow(videoFrame), hold on, title('Detected features');
plot(points);
```

#### **3.1.4 Track Features:**

Matlab uses `vision.PointTracker` System object to track various feature points. The point tracker find the corresponding point in the current frame. Then the `estimateGeometricTransform` function is used to estimate the translation, rotation, and scale between the old points and the new points. This transformation is applied to the bounding box around the face.

### 3.1.5 Track faces:

After detecting feature points matlab code tracks all points from frame to frame. `estimateGeometricTransform` function is used to calculate the motion of the face.

```

% Make a copy of the points to be used for computing the geometric
% transformation between the points in the previous and the current
frames
oldPoints = points;

while ~isDone(videoFileReader)
    % get the next frame
    videoFrame = step(videoFileReader);

    % Track the points. Note that some points may be lost.
    [points, isFound] = step(pointTracker, videoFrame);
    visiblePoints = points(isFound, :);
    oldInliers = oldPoints(isFound, :);

    if size(visiblePoints, 1) >= 2 % need at least 2 points

        % Estimate the geometric transformation between the old
points
        % and the new points and eliminate outliers
        [xform, oldInliers, visiblePoints] =
estimateGeometricTransform(...
        oldInliers, visiblePoints, 'similarity', 'MaxDistance',
4);

        % Apply the transformation to the bounding box points
bboxPoints = transformPointsForward(xform, bboxPoints);

        % Insert a bounding box around the object being tracked
bboxPolygon = reshape(bboxPoints', 1, []);
        videoFrame = insertShape(videoFrame, 'Polygon',
bboxPolygon, ...
        'LineWidth', 2);

        % Display tracked points
        videoFrame = insertMarker(videoFrame, visiblePoints, '+',
...
        'Color', 'white');

        % Reset the points
        oldPoints = visiblePoints;
        setPoints(pointTracker, oldPoints);
    end

    % Display the annotated video frame using the video player
object
    step(videoPlayer, videoFrame);
end

% Clean up
release(videoFileReader);
release(videoPlayer);
release(pointTracker);

```

### 3.1.6 Store the tracked faces to database:

After tracking faces different faces are stored in to the database for further recognition process. In our project we store 10 various faces of a single person in to the database. imwrite function is used for storing faces to a image database.

```

for i = 1:size(bbox,1)
J= imcrop(videoFrame,bbox);
I=imresize(J, [292,376]);
x=rand(1,1);
conct=strcat('data', num2str(x));
%disp(conct);
baseFileName = sprintf('%s %d.jpg', conct, frameCount);
    fullFileName = fullfile(pathName, baseFileName);
%save cropped faces in database folder
filename = ['F:\matlab\bin\face_database\' num2str(i+(size(bbox,1)))
'.jpg'];
% subplot(2,2,i);
    imwrite(J,fullFileName);

end

```

## 3.2 Testing process:

### 3.2.1 Real time Image capturing:

Image capturing is done by the use of webcam where user have to stand in front of camera .captured images are send to matlab for detecting and tracking image features.

### 3.2.2 Detect faces:

Faces which are captured from webcam are detected using viola jones algorithm using cascade object detector class. Here faces are cropped along with their bounding boxes and pass to features extraction process.

### 3.2.3 Extract good features:

After detecting faces KLT algorithm tracks a set of feature point across the video frames. Following are the matlab code for extract good features.

### 3.2.4 Track Features:

Matlab uses `vision.PointTracker` System object to track various feature points. The point tracker find the corresponding point in the current frame. Then the `estimateGeometricTransform` function is used to estimate the translation, rotation, and scale between the old points and the new points. This transformation is applied to the bounding box around the face.

### 3.2.5 Track faces:

After detecting feature points matlab code tracks all points from frame to frame. `estimateGeometricTransform` function is used to calculate the motion of the face.

### 3.2.5 Load database:

All faces from face database are loaded for access and results are used for Comparison.

```

srcFiles = dir('face_database\*.jpg'); % the folder in which ur images exists
for i = 1 : length(srcFiles)
    filename = strcat('face_database\' ,srcFiles(i).name);
    I = imread(filename);
    % figure, imshow(I);
    subplot(2,2,i);
    imshow(I);
end

```

### 3.2.6 Comparison of faces:

Test image is compared with images from the face database. Images with the closest features are matched. Comparison of various features of test images and trained image are done by Wiener filtering techniques. Faces are recognised from the matching result.

```

if frameCount==10
for i = 1:size(bbox,1)
J= imcrop(videoFrame,bbox(i,:));
I=imresize(J,[292,376]);

    %imwrite(I);
    %imshow(I);
    A = rgb2gray(I);
%figure;
%imshow(boxImage);
for j = 1 : length(srcFiles)
%try
    filename = strcat('face_database\',srcFiles(j).name);
    J = imread(filename);
    M = rgb2gray(J);
%figure,imshow(A);
%title('after gray conversion A');
%figure,imshow(M);
%title('after gray conversion M');

```

```

J=wiener2(A,[5 5]);
%title('image after wiener filtering A');
%figure,imshow(J);

J1=wiener2(M,[5 5]);
%title('image after wiener filtering M');
%figure,imshow(J1);

BW1=edge(J,'canny');
%figure,imshow(BW1);
%title('image after edge detection A');

BW2=edge(J1,'canny');
%figure,imshow(BW2);
%title('image after edge detection M');

OUTPUT_MESSAGE = 'almost same x-ray images ';
OUTPUT_MESSAGE2 = ' x-ray images not matching ';
matched_data = 0;
white_points1 = 0;
white_points2 = 0;
black_points = 0;
x=0;
y=0;
l=0;
m=0;
time=0;
for a = 1:1:256
    for b = 1:1:256
        if(BW1(a,b)==1)
            white_points1 = white_points1+1;
        else
            black_points = black_points+1;
        end
    end
end
end

```



```

display(white_points1);
display(white_points2);

%total_data = white_points;
total_matched_percentage = (white_points1/white_points2)*100;
display((total_matched_percentage));

if(total_matched_percentage >= 85)

    time='face matches 85%';
    set(handles.text6, 'String', 'Congrates..face matches')
display(time);
a = rand;
url =
'http://localhost/ebanking_phd/pages/examples/verify_otp.php?suc=ret';
web(url, '-browser');
    break;
elseif(total_matched_percentage >= 50 && total_matched_percentage <= 85)

    time='face matches 40%';
    set(handles.text6, 'String', 'Congrates..face matches')
display(time);
url =
'http://localhost/ebanking_phd/pages/examples/verify_otp.php?suc=ret';
web(url, '-browser');
    break;
else

    time='face notmatches';
display(time);
set(handles.text6, 'String', 'Sorry.....face not matches..Try
to login again')
url =
'http://localhost/ebanking_phd/pages/examples/verify_otp.php?suc=ret_not'
;
    web(url, '-browser');

end
%catch
disp('face not matching');
%disp( frameCount);
%end
end

end
end
end

clear cam;
release(videoPlayer);
release(pointTracker);
release(faceDetector);

```

There is another area login through face recognition. Start camera button is used for open webcam (Fig. 3) for recognition process

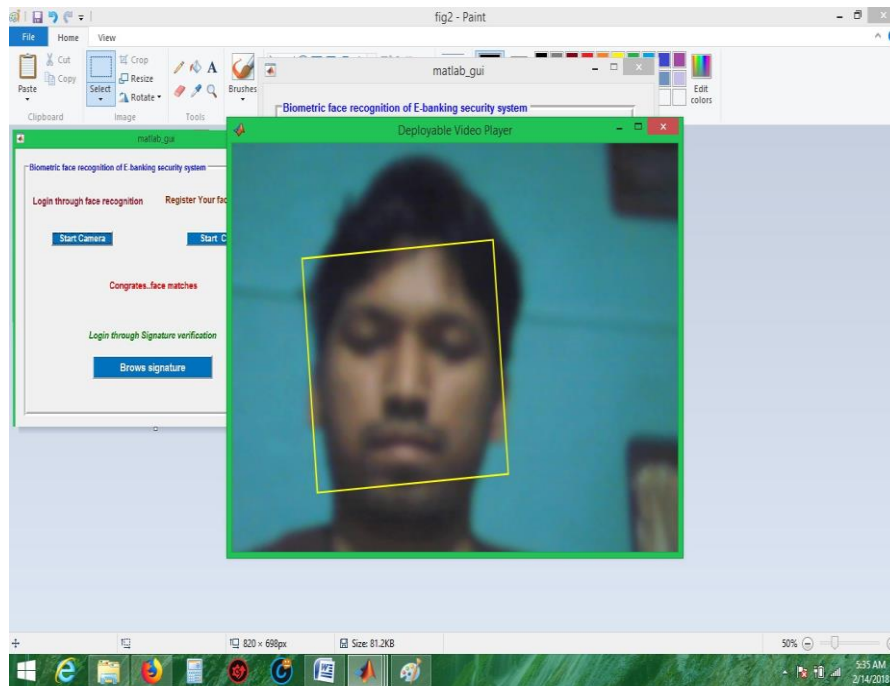


Fig.3 Tracking face from webcam

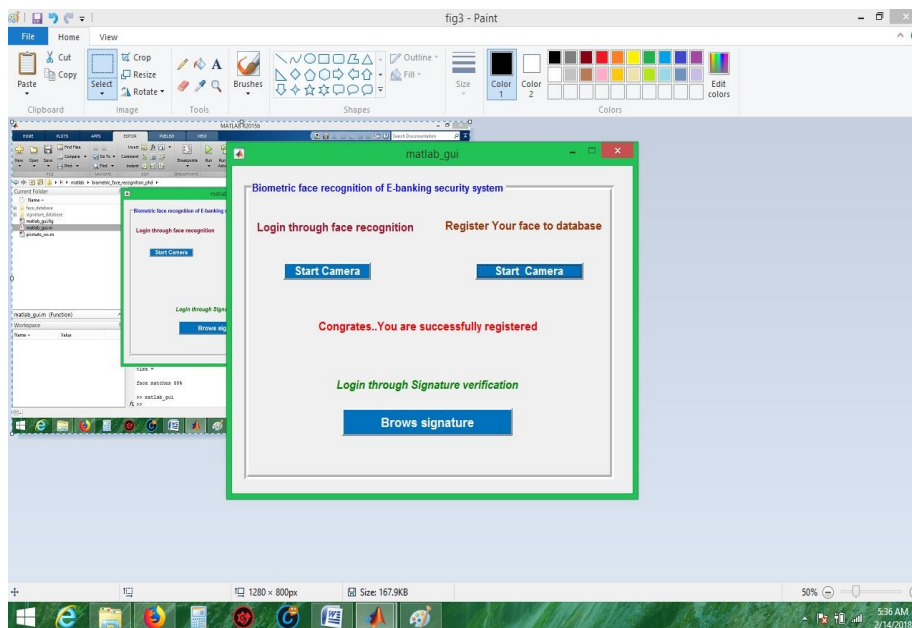


Fig. 4: Successful message after storing of faces to face database

Fig. 4 indicates trained faces are stored to database using webcam. A successful message will "Congrates....You are successfully registered" come after storing faces to database.

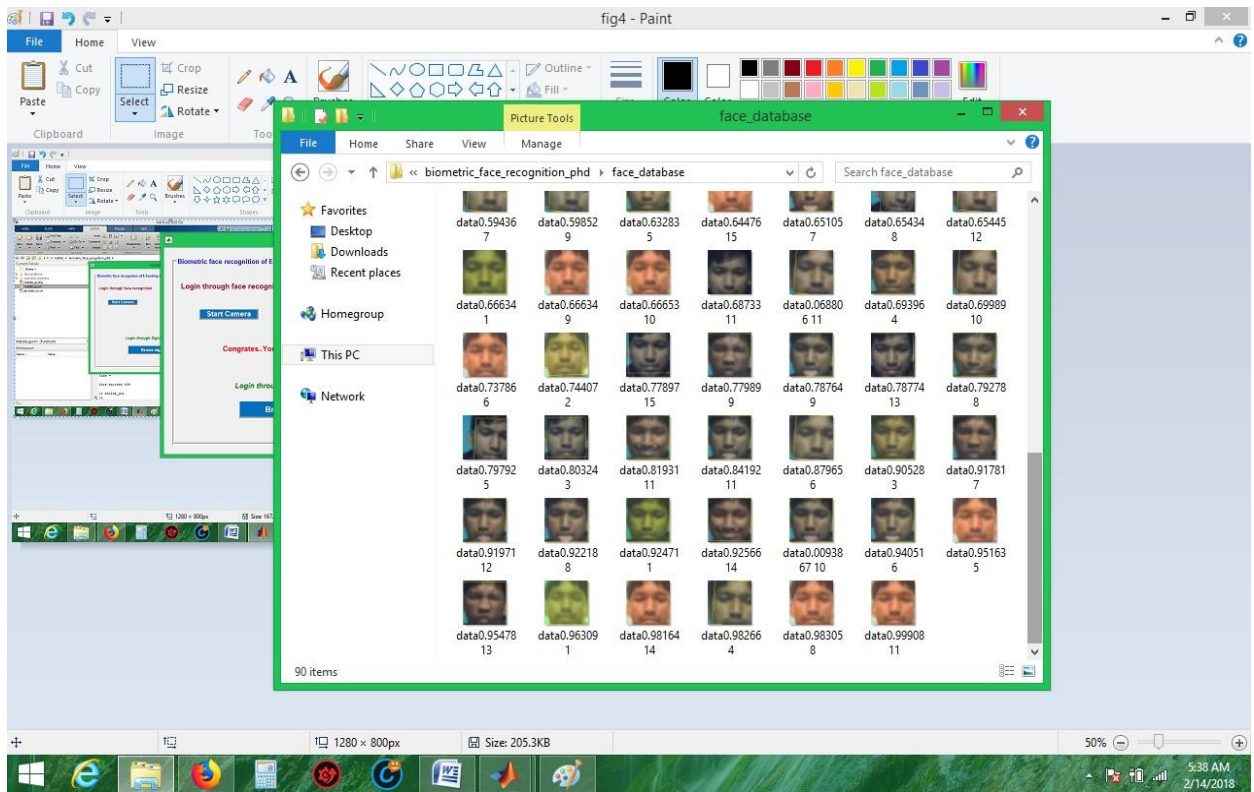


Fig. 5: Face database

Fig. 5 shows trained images in database. All the images files of the database can be loaded for recognition process.

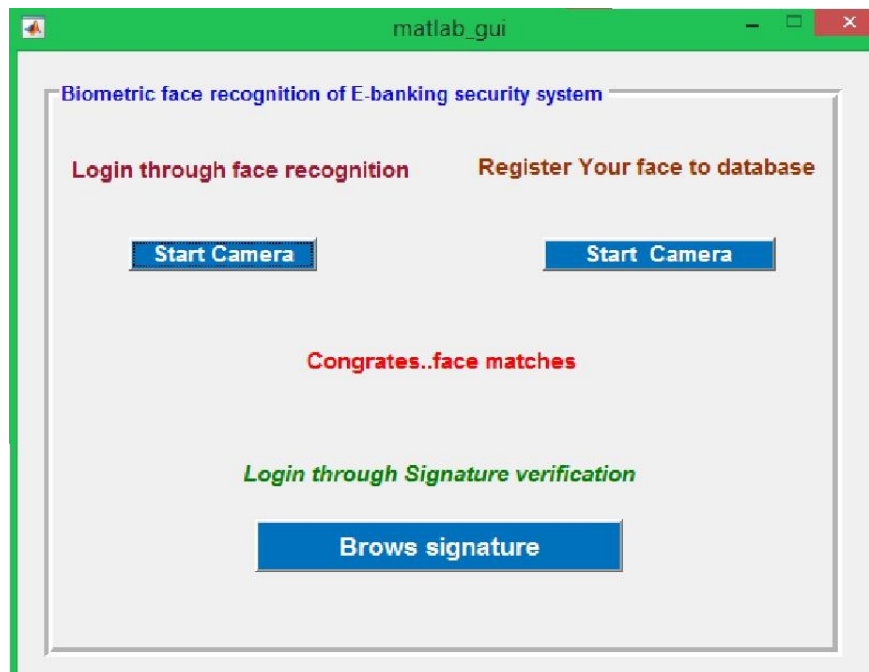


Fig. 6: Successful message after face recognition

Fig. 6 Shows the successful recognition message “congrates...face matches” if various features points of both trained and test images matched after taking real time test face of a person using webcam.

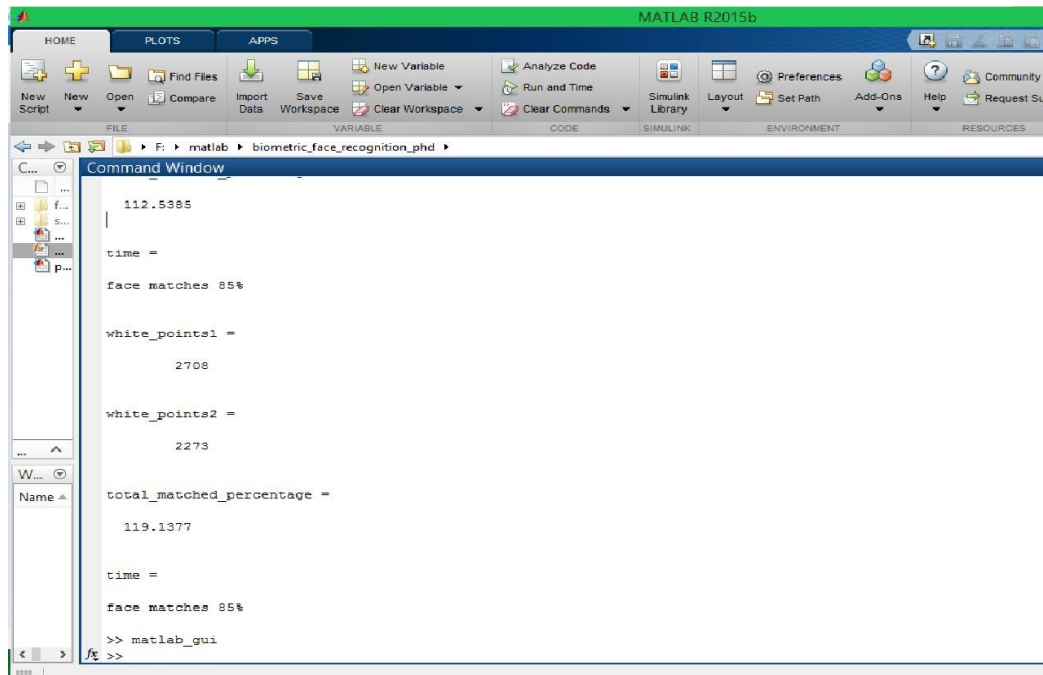


Fig. 7: Backend matlab code in editor

Fig. 7 shows matching percentage of trained and test images in matlab environment. In this figure we are trying to display matlab code that calculates matching percentage of two images.

## 6. CONCLUSION

Our aim is to implement a face detection and recognition system using KLT algorithm and Weiner filtering. We used these two algorithms because both the algorithm finds the best alignment for each possible location in the search space and minimize the overall mean square error in the process of inverse filtering and noise smoothing Frequency domain respectively. These are very powerful techniques and can be used in modern day security systems. This project can be extended in future for the fields like e-banking sector, Criminal investigation system, social networking site, national security system, etc.

## REFERENCES

- [1] Shamla Mantri, Kalpana Bapat , ‘Nural network based face recognition using matlab’,IJCSMC, Feb 2011, Vol 1.
- [2] Riresh Boda , M.Jasmine Pemeena Priyadarshini, ‘Face detection and tracking using KLT and Viola Jones’
- [3] Viola, Paul A. and Jones, Michael J. "Rapid Object Detection using a Boosted Cascade of Simple Features", IEEE CVPR, 2001.
- [4] Carlo Tomasi and Takeo Kanade. Detection and Tracking of Point Features. Carnegie Mellon Technical Report CMU-CS-91-132, 1991.
- [5] Ponya Bagherpour , Seyed Ali Cheraghi , Musa bin Mohd Mokji, ‘International Neural network society Winter Conference(INNS-WC2012)’, 2012

- [6] Wiener, Norbert (1949). *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. New York: Wiley. ISBN 0-262-73005-7.
- [7] Neeraj Kumar ,karambir , Kalyan Singh,"International journal for electronic eng",3(2), ISSN : 0973-7383,2011
- [8] MARAPAREDDY. R," International journal for electrical,electronic and data communication",ISSN:2320- 2084,Vol-5,Issue-8,2017
- [9] Ekta Thaman, RamanpreetKour,"Face detection using using spatio-temporal segmentation and tracking for video security",IJEDR,2015,vol 3,issue 3,ISSN:2321-9939
- [10] Prof. Rushabh Shah ,Aalisha Sheth,Biyanta Shah,Ayushi Shah ,Dr. Priyanka Sharma,"Comparative study and analysis of face detection algorithms",2015, International Journal of Advanced Technology in Engineering and Science,Vol-3,ISSN:2348-7550.
- [11] Ritesh Boda and M. Jasmine Pemeena Priyadarsini," Face detectionand tracking using KLT and Viola Jones ",2016,Vol 11,ISSN :1819-6608\
- [12] P.J Leo Evenss, Jennings Mcenroe .S, A.Prabhu Chakkaravarthy ," Dynamic face recognition and tracking system using machine learning in matlab and big data", NTERNATIONAL JOURNAL ON SMART SENSING AND INTELLIGENT SYSTEMS SPECIAL I,2017,Issue September
- [13] Kapil, Anurag Singh," Secured Transmission of Video by Advantageous KLT Algorithm",2015,International journal of innovative research in computer and communicative Engineering,Vol 3,Issue 7,2015
- [14] Raajan.P,Muthuselvi.S,Agnes Saleema.A,"An adaptive image enhancement using weiner filtering with compression and segmentation",International journal for computer application,ISSN:0975-8887,2015.
- [15] Prodip Biswas,Abu Sufian sarkar,Muhammed Mynuddin,"Deblurring images using a Weiner filter", International journal for computer application, ISSN:0975-8887,Vol 109,2015