Original Research Article

Performance Assessment of Principal Component Analysis and Kernel Principal Component Analysis using TOAM Database

Abstract

Face Recognition algorithms can be classified into Appearance-based (Linear and Non-Linear Appearance-based) and Model-based Algorithms. Principal Component Analysis (PCA) is an example of Linear Appearance-based which performs a linear dimension reduction while Kernel Principal Component Analysis (KPCA) is an example of non-linear appearance methods. The study focuses on the performance assessment of PCA and KPCA face recognition techniques. The assessment is carried out base on computational time using Testing Time and recognition accuracy on created database identified as TOAM database. The created database is mainly for this research purpose and it contains 120 face images of 40 persons frontal faces with 3 images of each individual under different lighting, facial expressions, occulations, environment and time. ~~80 images was used for training while 40 is use for testing the images to test the recognition accuracy and computational time of the system.~~ The findings reveal an Average Testing Time of 1.5475 seconds for PCA and 67.0929 seconds for KPCA indicating a longer Computational time for KPCA than PCA. It also reveals that PCA has 72.5% performance recognition accuracy while KPCA has 80.0% performance recognition accuracy indicating that KPCA outperforms the PCA in terms of recognition accuracy.

Keywords: Kernel Principal Component Analysis, Principal Component Analysis, Performance, Face recognition, Computational time.

1. Introduction

Rise in criminals in the world especially in Nigeria is as a result of poor identification and verification of citizenry and immigrants. Several methods of identification has been in existence from time immemorial such as tribal marks, names, intonations and so on. Also, passwords (knowledge-based scheme) and ID cards (token-based schemes) have been used to validate the identity of an individual intending to access the services offered by an application such as online transaction. Establishing the identity of an individual is of paramount importance in our highly networked society (Ross, 2007). All the listed methods for user authentication have several limitations for example tribal mark is tagged as crude and defacing, simple passwords can be revealed or easily guessed by unauthorized users, complex passwords can be difficult to recollect for a legitimate user, ID cards can be misplaced, forged or stolen. In order to have a strong and better mode of identification and verification biometric is adopted. Biometric is highly reliable, cannot be easily faked, provides strong authentication and user convenience. Among the mostly used biometric features are the face, fingerprint, voice, Deoxyribonucleic Acid (DNA), retina, and the iris.

Face Recognition (FR) is a Visual Biometric**.** It utilizes distinctive features of the face to authenticate users. The discipline that cut across FR includes computer vision, neural network, pattern recognition and image processing (Sushma, Sarita, & Rakesh, 2011). Major benefits of facial recognition are that it is non-intrusive, hands-free, continuous and accepted by most users (Bolle, Connell, Pankanti, Ratha & Senior, 2004). Those major identified challenges hindering face recognition system are illumination, ageing, camera quality, the emotional perception and occlusion. Also, Fagbola, Olabiyisi, Egbetola, and Oloyede, (2017) said face recognition from unconstrained scenes has been a subject of debate among researchers as a result of the massive influx of video surveillance system (VSS) and other ubiquitous hand-held video capturing devices.

2. Related Literature

2.1 Related Literature

Support Vector Machine-based algorithm is judged with a principal component analysis (PCA) based algorithm on a difficult set of images from the FERET database (Philips, 1999). Performance was measured for both verification and identification setups. The identification performance for SVM is 77% to 78% while PCA is 54%. PCA has 13% verification as against SVM 7%.

~~In 2003,~~ Draper, Baek, Bartlett and Beveridge, (2003) assess Principal Component Analysis (PCA) and Independent Component Analysis (ICA) face recognition system. The work explores the space of PCA and ICA comparisons with four different distance measures on two tasks (facial identity and facial expression). In all cases, PCA performs well but not as well as ICA.

Adedeji, Omidiora, Olabiyisi and Adigun, (2012) used recognition accuracy, total training time and average recognition time as performance metrics in evaluation of Optimised PCA (OPCA) and Projection Combined PCA ((PC)2A) techniques. The outcomes of assessment between both methods based on black faces showed that OPCA and (PC)2A provided recognition accuracies between 96% to 64% and between 95% to 60% respectively. The results showed that OPCA required more training time than (PC)2A but it acquired a longer time to recognize images with (PC)2A than OPCA. General results shown that OPCA performed better than (PC)2A.

Aluko, Omidiora, Adetunji and Odeniyi, (2015) perform their experiment on three selected PCA-based techniques for face recognition. Principal Component Analysis (PCA), Binary Principal Component Analysis (BPCA), and Principal Component Analysis – Artificial Neural Network (PCA-ANN). The result showed that PCA, BPCA and PCA-ANN had recognition rates of 91%, 86% and 94% with recognition time of 5.2 seconds, 5.5 seconds and 140.5 seconds when 75 eigenvectors were selected. The occurrence assessment of the three PCA-based systems revealed that PCA – ANN techniques gave the best recognition rate of 94% with a trade-off in recognition time.

2.2 Feature Extraction Algorithms

Several feature extraction algorithms are in existence, most of them are used in areas other than face recognition. Some of well-known feature extraction algorithms are Principal Component Analysis (PCA), Kernel PCA, Linear Discriminant Analysis (LDA), Independent Component Analysis (ICA), Active Shape Models (ASM), Discrete Cosine Transform (DCT), Neural Network based methods, Semi-supervised Discriminant Analysis and so on. Face recognition algorithms can be classified as either geometry based or template based algorithms (Torres, 2004). Guo, Zhang and Li (2001) in their own case classified FB algorithm as Appearance based and Model based Algorithms. They furthered group Appearance based algorithm into Linear and Non-linear appearance based while the

Model based can be 2D or 3D. Linear appearance-based methods perform a linear dimension reduction examples of this approach are PCA, LDA or ICA while non-linear appearance methods are more knotty, Kernel PCA (KPCA) is an example. One example each of the linear and non-linear will be used for the assessment to determine the performance of linear over non-linear feature extraction algorithms.

2.3 Principal Component Analysis

The actual target of PCA is the dimensionality reduction. It is a scientific gismo for achieving dimensionality reduction in face recognition system. It is also known as Eigenspace projection or Karhumen-Loeve transformation (Turk & Pentland, 1991). The main idea of using PCA for face recognition is to express the large 1-D vector of pixels constructed from 2-D facial image into the compact principal components of the feature space (Aluko, Omidiora, Adetunji & Odeniyi, 2015). Karl Pearson invented PCA in 1901, but proposed for pattern recognition 64 years later. Finally, it was applied to face representation and recognition in the early 90's (Turk & Pentland, 1991).

Usually the mean x is extracted from the data, so that PCA is equivalent to Karhunen-Loeve Transform (KLT). So, let $x_n x_m$ be the data matrix where x_1, ..., x_m are the image vectors (vector columns) and n is the number of pixels per image. The KLT basis is obtained by solving the eigenvalue problem where C_x is the covariance matrix of the data.

$$C_x = \phi\Lambda\phi^T$$

$$C_x = \frac{1}{m}\sum_{i=1}^{m} xixi^T$$

$\phi = [\O_1, \ldots ..\O_n]$ is the eigenvector matrix of C_x . Λ is a diagonal matrix, the eigenvalues $\lambda_1, \ldots, \lambda_n$ of C_x are located on its main diagonal. λ_i is the variance of the data projected on $\O_i$.

2.3.1 Some function in Principal Component Analysis (PCA) Algorithm

```
function [m, A, Eigenfaces] = EigenfaceCore(T)
% Use Principal Component Analysis (PCA) to determine the most
% discriminating features between images of faces.

%%% Calculating the mean image
m = mean(T,2);
% Computing the average face image m = (1/P)*sum(Tj's)    (j = 1 : P)
Train_Number = size(T,2);

%%% Calculating the deviation of each image from mean image
```

```
A = [];
for i = 1 : Train_Number
    temp = double(T(:,i)) - m;
% Computing the difference image for each image in the training set Ai = Ti - m
    A = [A temp]; % Merging all centered images
end

%%% Snapshot method of Eigenface methods

L = A'*A;
% L is the surrogate of covariance matrix C=A*A'.
[V D] = eig(L);
% Diagonal elements of D are the eigenvalues for both L=A'*A and C=A*A'.

%%% Sorting and eliminating eigenvalues
L_eig_vec = [];
for i = 1 : size(V,2)
    if( D(i,i)>1 )
        L_eig_vec = [L_eig_vec V(:,i)];
    end
end

%%% Calculating the eigenvectors of covariance matrix 'C'
Eigenfaces = A * L_eig_vec; % A: centered image vectors
```

2.4 Kernel PCA

Kernel Principal Component Analysis is the nonlinear form of PCA, which is good in accomplishment of complicated spatial structure of high-dimensional features. Its basic methodology is to apply a non-linear mapping to the input ($\Psi(x)$: $R^N \rightarrow R^L$) and then solve a linear PCA in the resulting feature subspace. The mapping of $\Psi(x)$ is made implicitly using kernel functions

$$k(x_i , x_j) = (\Psi(x_i). \Psi(x_j))$$

where n the input space correspond to dot- products in the higher dimensional feature space.

2.4.1 Some function in Kernel Principal Component Analysis (KPCA) Algorithm

```
%%% Calculating the mean image
m = mean(T,2);
% Computing the average face image m = (1/P)*sum(Tj's)    (j = 1 : P)
Train_Number = size(T,2);

%%% Calculating the deviation of each image from mean image
A = [];
for i = 1 : Train_Number
    temp = double(T(:,i)) - m;
% Computing the difference image for each image in the training set Ai = Ti - m
    A = [A temp]; % Merging all centered images
end

%% Using the Gaussian Kernel to construct the Kernel K
```

```
% K(x,y) = exp(-(x-y)^2/2(sigma)^2)
% K is a symmetric Kernel
K = zeros(size(A,2),size(A,2));
for row = 1:size(A,2)
    for col = 1:row
        temp = sum(((A(:,row) - A(:,col)).^2));
        K(row,col) = exp(-temp./(2*(0.26*size(T,1))^2)); % sigma = 1
    end
end
    K = K + K';
% Dividing the diagonal element by 2 since it has been added to itself
for row = 1:size(T,2)
    K(row,row) = K(row,row)/2;
end
one_mat = ones(size(K));
K_center = K - one_mat*K - K*one_mat + one_mat*K*one_mat;

%%%%K_center is inner dot product matrix in feature space matrix vector
[V,D] = eig(K_center);
evecs = V;
evals = real(diag(D));
for i=1:Train_Number,
  evecs(:,i) = evecs(:,i)/(sqrt(evals(i)));%dividing eigen vector by sqr root of
corresponding eig values
end
%%% Calculating the eigenvectors of covariance matrix 'C'
Eigenfaces = A * evecs;   % A: centered image vectors
ProjectedImages = [];
Train_Number = size(Eigenfaces,2);
for i = 1 : Train_Number
    temp = Eigenfaces'* A(:,i); % Projection of centered images into facespace
      ProjectedImages = [ProjectedImages temp];
      ProjectedImages1 =imresize(ProjectedImages ,[40,40]);
      set(handles.text3,'string','Aggregating Features Vectors');
      axes(handles.axes1);
      imshow(ProjectedImages1 );
pause(0.1)

end
```

3. Methodology

3.1 Database Setup

Adedeji, Omidiora, Olabiyisi and Adigun (2012) pointed out that the higher the resolution of cropped image, the more time it takes to train the database and that the total training time also increases with increase in the number of training images per person. Putting this in mind a Face Recognition System Database containing 120 facial images was created purposely for the research work and identified it as TOAM DATABASE. 40 persons frontal faces with 3 images of each individual under different lighting, facial expressions, occulations, environment and time was captured into the database . The captured images go through geometric normalisation in order to get better output. 80 images were used for training while 40 were used for testing. The images in TOAM database were transformed into gray colour in order to make suitable for the FR system because two-dimensional arrays are required by majority of the face recognition algorithms for analysis.

Figure 1: Some of the images used for training TOAM database

3.2 System Design

MATLAB R2015a was used to implement PCA and KPCA algorithms on Intel(R) Celeron (R) CPU with 1.60GHz Processor speed. The experiment was with total of 120 facial images, out of which 80 images were used as shown in Table 1. At the end of the experiment, recognized index in Database and Testing time were used as performance metrics to determine the computational time and recognition accuracy. The system consists of number of modules: image acquisition, Feature extraction, recognition accuracy. PCA and KPCA are the two dimensionality reduction algorithms used in the feature extraction in face recognition and Euclidean distance was used for classification technique.

Table 1: Analysis of the Data used for the in TOAM database

Number of persons	40
Number of sample per persons	3
Number of Total sample	120
Number of Training set	80
Number of Testing sample	40

3.2.1 Image acquisition

Images were captured with camera for the setup database identified as “TOAM Database”. The images captured went through geometric normalisation in order to get better output. 80 images designated for training while 40 will be used for testing as shown in Table 1. The images in the database were transformed into gray colour.

3.2.2 Feature Extraction

Feature extraction is the act of obtaining momentous evidence from a face image. It process must be efficient in terms of computing time and memory usage. Dimensionality reduction and feature selection are the main stages in Feature extraction.

3.2.3 Euclidean Distance

Euclidean distance or Euclidean metric is used as a classifier for incoming test data. It is an ordinary straight line distance between two points in the plane e.g Practical Machine Learning. It scrutinizes the root of square difference between matches of a pair of objects.

$$\mathrm{di} = \sqrt{\sum_{n-1}^{n}\left(x_{ik} - x_{jk}\right)^2} \qquad \ldots.1$$

4. Result and Discussion

Table 2: Analysis of Computational Time for both PCA and KPCA using the same sample images

IMAGE	PCA TESTING TIME (Seconds)	KPCA TESTING TIME (Seconds)
1	1.6323	67.26878
2	1.5212	67.18138
3	1.5536	67.17698
4	1.5065	67.09458
5	1.5333	66.99618
6	1.5881	67.11408
7	1.5204	67.01988
8	1.5088	67.15838
9	1.5432	67.07368
10	1.5360	67.18768
11	1.5242	67.15048

12	1.5538	67.42578
13	1.4931	66.99218
14	1.5349	67.21768
15	1.5439	67.21188
16	1.5497	67.29888
17	1.5257	67.10158
18	1.5435	67.27998
19	1.5237	66.99628
20	1.5453	67.07438
21	1.5414	67.22328
22	1.5374	67.14088
23	1.5468	67.23318
24	1.5342	67.24188
25	1.5396	67.35968
26	1.5572	67.12928
27	1.5809	67.69498
28	1.5747	67.25228
29	1.5240	68.97878
30	1.5565	68.36638
31	1.5929	66.90718
32	1.5386	67.06158
33	1.5257	66.89868
34	1.5680	66.99938
35	1.5701	66.95388
36	1.5853	67.08668
37	1.5665	66.91238
38	1.5676	68.77168
39	1.5568	68.33738
40	1.5560	67.49338
Total	**61.9014**	**2683.925**
Average	**1.5475**	**67.3016**

Table 2 and Figure 2 shown the variation of Testing Time (TT) used by both PCA and KPCA on each image. It was deduced that each of the image TT used by PCA is far lesser than those of KPCA. The reserch reveals an Average Testing Time of 1.5475 seconds for PCA and 67.3016 seconds for KPCA. The asssessement is that the Computational Time of KPCA is more than that of PCA.

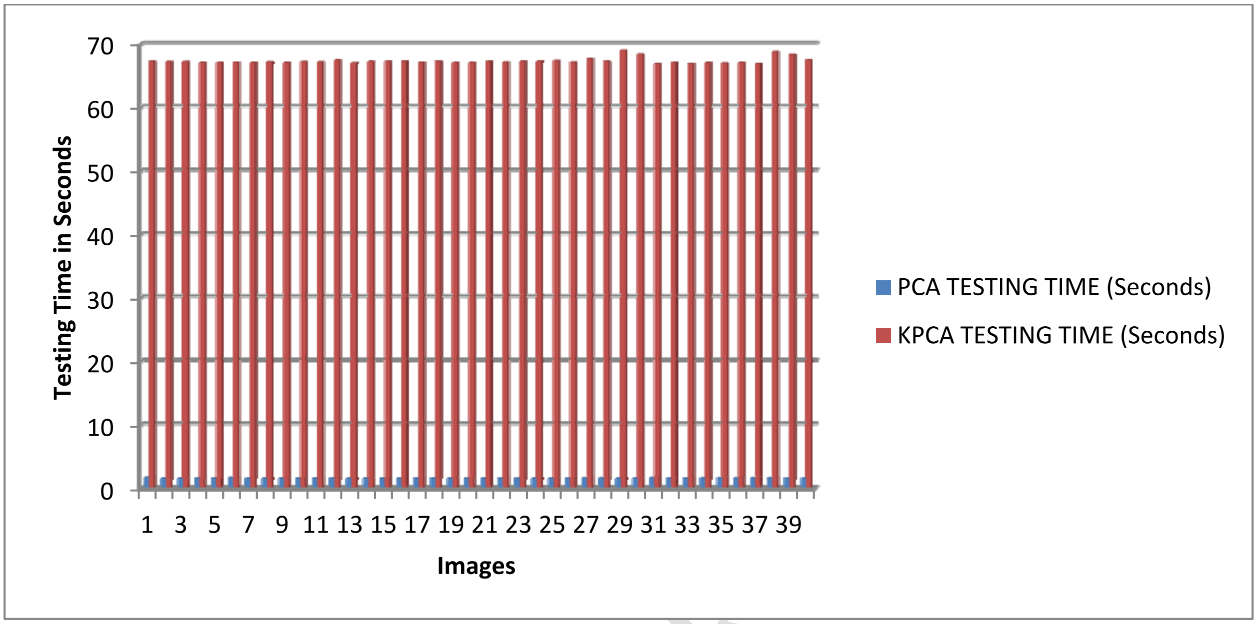

Figure 2: Testing Time for PCA and KPCA

Table 3: Analysis of both PCA and KPCA Recognition Performance using the same sample image

IMAGE	PCA RECOGNIZED INDEX IN DATABASE	PCA RECOGNIZED	KPCA RECOGNIZED INDEX IN DATABASE	KPCA RECOGNIZED
1	75.jpg	NO	75.jpg	NO
2	73.jpg	NO	73.jpg	NO
3	5.jpg	YES	5.jpg	YES
4	8.jpg	YES	8.jpg	YES
5	9.jpg	NO	9.jpg	NO
6	12.jpg	YES	12.jpg	YES
7	13.jpg	YES	13.jpg	YES
8	66.jpg	YES	66.jpg	YES
9	20.jpg	YES	20.jpg	YES
10	20.jpg	YES	20.jpg	YES
11	23.jpg	YES	23.jpg	YES

12	23.jpg	YES	23.jpg	YES
13	25.jpg	NO	27.jpg	YES
14	26.jpg	NO	27.jpg	YES
15	32.jpg	YES	32.jpg	YES
16	29.jpg	YES	29.jpg	YES
17	33.jpg	YES	33.jpg	YES
18	17.jpg	NO	17.jpg	NO
19	40.jpg	YES	40.jpg	YES
20	39.jpg	YES	39.jpg	YES
21	42.jpg	YES	42.jpg	YES
22	17.jpg	NO	17.jpg	NO
23	44.jpg	NO	44.jpg	NO
24	36.jpg	NO	36.jpg	NO
25	52.jpg	YES	52.jpg	YES
26	50.jpg	YES	50.jpg	YES
27	54.jpg	YES	54.jpg	YES
28	54.jpg	YES	54.jpg	YES
29	59.jpg	YES	59.jpg	YES
30	59.jpg	YES	59.jpg	YES
31	62.jpg	YES	62.jpg	YES
32	65.jpg	NO	65.jpg	NO
33	66.jpg	YES	66.jpg	YES
34	67.jpg	YES	67.jpg	YES
35	72.jpg	YES	72.jpg	YES
36	72.jpg	YES	72.jpg	YES
37	76.jpg	YES	76.jpg	YES
38	75.jpg	YES	75.jpg	YES
39	78.jpg	YES	78.jpg	YES
40	9.jpg	NO	79.jpg	YES

Table 4: Summary of both PCA and KPCA Recognition Performance using the same sample image

	PCA RECOGNIZED	KPCA RECOGNIZED
Number of **YES**	29	32
Number of **NO**	11	08
Total	40	40
Percentage of Recognition Performance	**72.5%**	**80.0%**

The study of Table 4 clearly shows the analysis of both PCA and KPCA Recognition Performance Accuracy using the same sample image. PCA was unable to recognize images 1,2,5,13,14,18,22,23,24,32 and 40 while KPCA was unable to recognize images 1,2,5,18,22,23,24 and 32. It was revealed that image 13,14 and 40 which PCA recognized index in database were

25.jpg, 26.jpg and 9.jpg respectively were mismatched but which was properly recognized by KPCA as 27.jpg, 27.jpg and 79.jpg respectively. Table 4 and Figure 3 also revealed that PCA was able to Recognize 29 images while KPCA recognized 32 images. PCA has 72.5% performance recognition accuracy while KPCA has 80.0% performance recognition accuracy. The assessment is that KPCA performs better than PCA in terms of Performance Recognition accuracy.

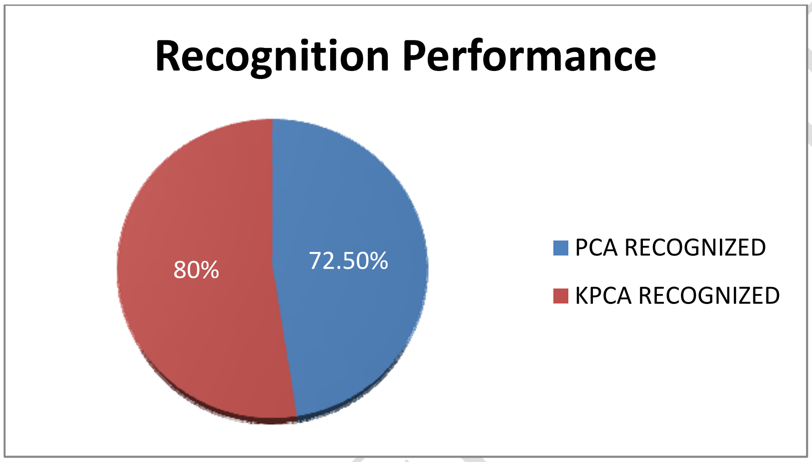

Figure 3: Performance Recognition Accuracy for PCA and KPCA

5. Conclusion

A brief background study of biometric and face recognition ~~feature extraction~~ algorithms were presented. This study assesses the ~~usefulness and~~ performance of both PCA and KPCA ~~to assess the~~ Computational time and ~~performance~~ face recognition accuracy. The experimental results shown an Average Testing Time of 1.5475 seconds for PCA and 67.0929 seconds for KPCA, it implies that it takes a longer Computational time for KPCA than PCA. However, the experiment revealed that PCA has 72.5% performance recognition accuracy while KPCA has 80.0% performance recognition accuracy, indicating that KPCA outperforms the PCA in terms of recognition accuracy. It should be noted that the results ~~got~~ were basically limited by configuration of the computer system used, resolution of the digital camera, different environmental conditions like illumination and different distances between the camera and every face. In summary PCA tradeoff recognition accuracy for testing time while KPCA tradeoff testing time for recognition accuracy.

References

Adedeji, O. T., Omidiora, E. O., Olabiyisi, S. O. and Adigun, A. A. (2012). Performance Evaluation of Optimised PCA and Projection Combined PCA methods in Facial Images. *Journal of Computations & Modelling.* 2(3): pp17-29. ISSN: 1792-7625 (print), 1792-8850 (online).

Aluko, J. O., Omidiora, E. O., Adetunji, A. B. and Odeniyi, O. A. (2015). Performance Evaluation Of Selected Principal Component Analysis-Based Techniques For Face Image Recognition. *International Journal of Scientific & Technology Research.* 4(01): ISSN 2277-8616 Biometrics", New York, Springer-Verlag.

Bolle, R.M., Connell, J.H., Pankanti, S., Ratha, N.K. and Senior, K. (2004): "A Guide to Biometrics". *European Signal Processing Conference (EUSIPCO)*, Poznan, Poland.

Draper B. A., Baek K., Bartlett M. S. and Beveridge J. R. (2003). Recognizing faces with PCA and ICA. *Computer Vision and Image Understanding*. 91(1-2): 115 – 137 Elsevier Science Inc. New York, NY, USA

Fagbola ,T. M., Olabiyisi ,S. O., Egbetola, F. I. and Oloyede A. (2017). "Review of Technical Approaches to Face Recognition in Unconstrained Scenes with Varying Pose and Illumination." *FUOYE Journal of Engineering and Technology*, 2(1): ISSN 2579-0625.

Guo, G. D., Zhang, H. J. and Li, S. Z. (2001). "Pairwise face recognition". In *Proceedings of 8th IEEE International Conference on Computer Vision*. Vancouver, Canada.

Phillips, P. J. (1999). Support vector machines applied to face recognition. *Advances in Neural Information Processing Systems.* View@papers.nips.cc: pp. 803-809.

Ross A. (2007). "An Introduction to Multibiometrics". *15th European Signal Processing Conference (EUSIPCO)*, Poznan, Poland.

Sushma, J., Sarita, S. B., and Rakesh, S. J. (2011). 3D face Recognition and Modelling System. *Journal of Global Research in Computer Science*, 2(7): 30-37.

Torres, L. (2004). Is there any hope for face recognition? *In Proc. of the 5th International Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS*, pp. 21-23. Lisboa, Portugal.

Turk, M. and Pentland, A. (1991). Eigenfaces for recognition, *Journal of Cognitive Neuroscience*, **3**(1), (1991), 71-86.