Journal of Medical Systems (2019) 43: 12
https://doi.org/10.1007/s10916-018-1128-x

IMAGE & SIGNAL PROCESSING

LS-GSNO and CWSNO Enhancement Processes Using PCA Algorithm with LOOCV of R-SM Technique for Effective Face Recognition Approach

N. Rathika [1] · P. Suresh [1] · N. Sathya [1]

Received: 6 October 2018 / Accepted: 22 November 2018 / Published online: 7 December 2018

Abstract
The eminence of image under test is identified with different methods of Face Recognition (FR) which results in failure due to rapid change in pixel intensity. The identification of similar face with inter class similarity is very difficult in imaging. The imaging technology faces difficult in the mounting of intra class variability because of accommodate, intra-class variability because of head pose, illumination conditions, expressions, facial accessories, aging effects and cartoon faces. In the earliest approach, gradient with Zernike momemts were used to regonize the faces, the performance is low to overcome this a new approach is introduced. Many features of FR are affected by the outcome and low occurrence of performance is observed which is applicable only for data sets that are smaller. The introduction of a new approach can overcome the above stated limitations. This paper describes a novel approach for LS enhancement technique using GSNO and CWSNO, and extracts the PCA features with three ways such as mean, median and mode which are then classified with MD classifier using LOOCV of R-SM to recognize the faces. The performance metrics is also computed and compared. Performance metrics of the proposed approach and the current approach are computed and compared. Thus, the suggested method is useful for increasing the visibility of facial recognition, and overcoming a pose, similarity and illumination problem, which provides a more accurate investigation of the required recognition procedures.

Keywords LS-GSNO and CWSNO · Features extraction · Classifiers · Performance metrics

Introduction

The human body plays a vital role in image processing. The lobes of human brain and its perceptive abilities, cognition and consciousness, face recognition analysis in human being is the most trends among researchers were discussed in recent years [1] and it is difficult to understand the exact nature of all organs. The brain's temporal lobe has partial responsibility in understanding the known ability. The temporal lobe of human brain damage consequences the ability to recognize face of respective person. Due to little damage in temporal lobe region leads to loss in face recognition ability and it is called as prosopagnosia. Such persons found it difficult to recognize a person by sight but were able to recognize a person by voice [2].

Theoretical approach

The idea of Eigen images consideration for face recognition is formulated for 2-D recognition problem which is done by assumption [9]. This is because the faces will be mostly upright and frontal, [16] and the detailed 3-D information about the face is not required which reduces complexity in facial recognition. Eigen images were introduced before the facial recognition method [17] and it is observed from literature- that the recognition deals with local and intuitive features, such as the distance between eyes, ears and similar other features [18].

This article is part of the Topical Collection on *Image & Signal Processing*

✉ N. Rathika
rathika.111287@gmail.com; rathikapseee@gmail.com

P. Suresh
suresh3982@gmail.com

N. Sathya
sathyaeie@gmail.com

[1] Department of Electronics and Communication Engineering, P.S.N. College of Engineering and Technology, Tirunelveli, Tamil Nadu, India

Literature survey

The most inspired method which is based on information theory approach that can be applied to use intuitive feature is Eigen image method which is evident from the prior survey of research works [1]. The literature study also revealed that the Eigen image method is the best face recognition technique to identify and distinguish facial features through encoding of relevant information. The essential principle component includes the set of basic face images for the face image transformation process [19]. The representation of every image contained in the training sets can be done as the weighted linear combination for all basic faces [3]. The recognition of face is done by applied Zernike moment and its application is restricted for datasets that are smaller [5]. The number of Eigen images obtained is equal to the number of images available in the training sets. Principal Component Analysis (PCA) proves to be more efficient than several algorithm available [13, 15]. Garbor filters dimensionally increases with changes in the scale [12].

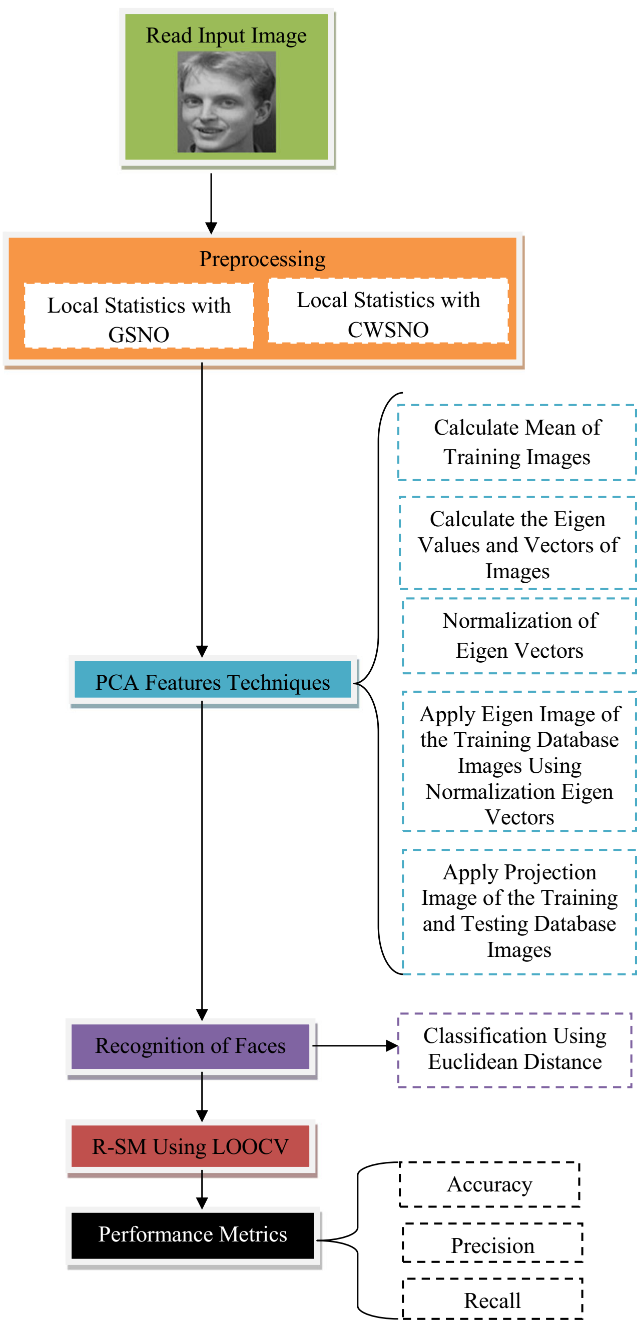


Fig. 1 Flow diagram of the proposed system of classification phase

Methodology

The proposed system involves following processes: (a) Classification Phase (b) Comparison Phase. These techniques are explained in the following sections.The flow

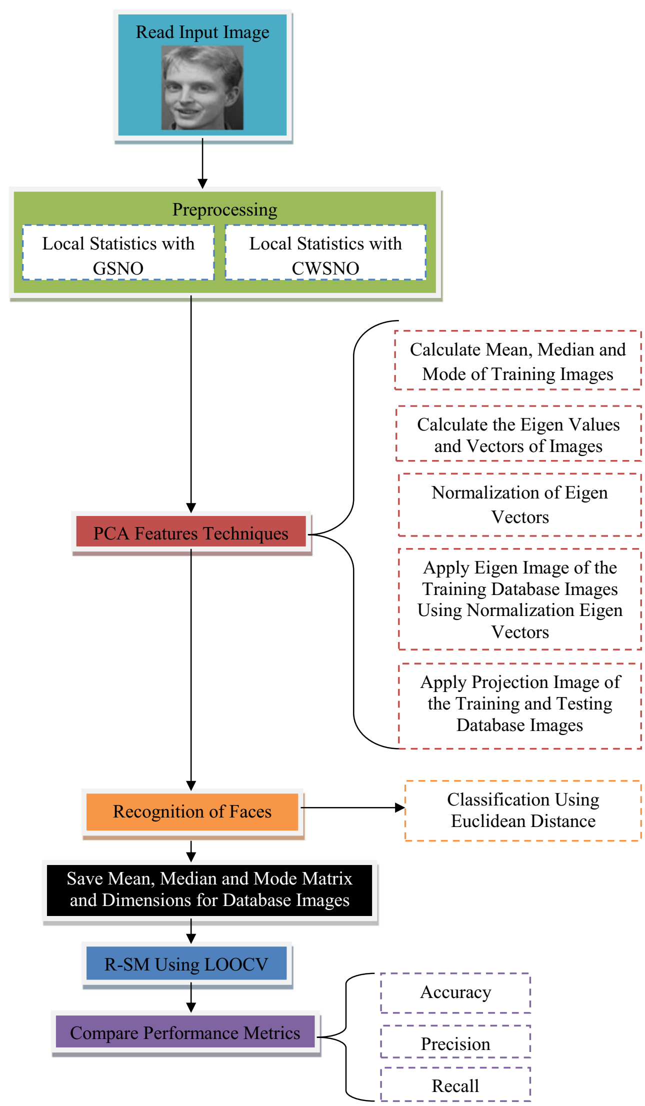


Fig. 2 Flow diagram of the proposed system of comparison phase

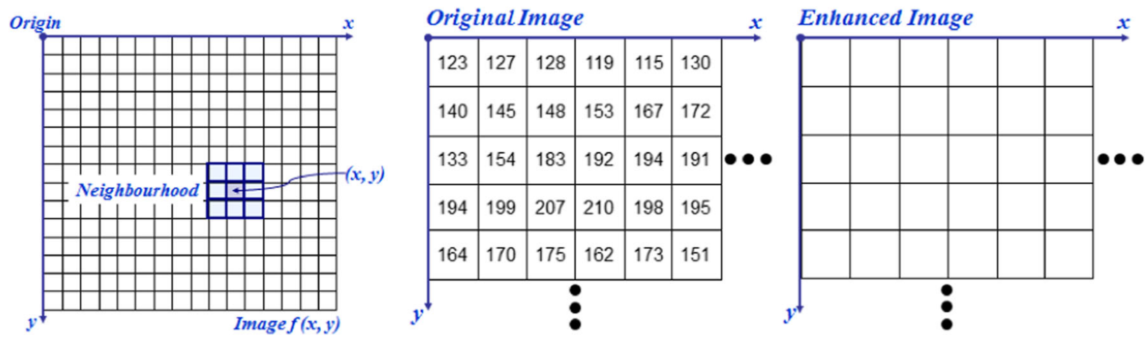


Fig. 3 Neighbourhood processing operation

diagram of the proposed system of classification phase as shown in the Fig. 1. The flow diagram of the proposed system of comparison phase as shown in the Fig. 2.

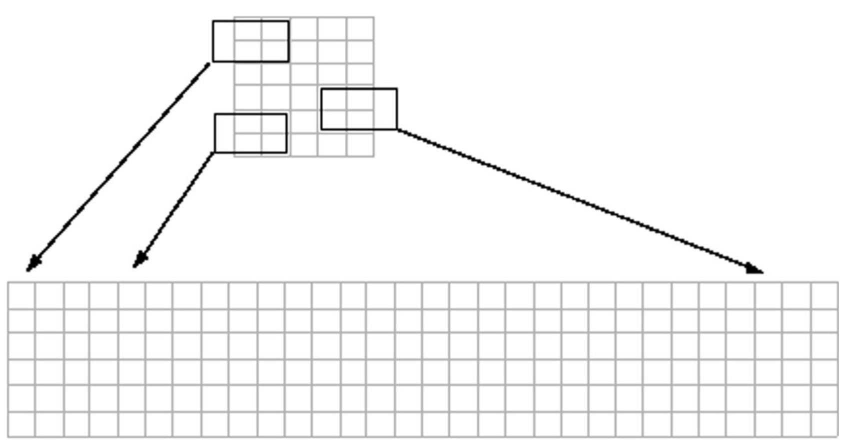
The classification phase consists of pre-processing using LS-GSNO and CWSNO, PCA feature extraction are obtained. The estimation of image quality base and intensity histogram is done using GSNO and CWSNO algorithm from which the best transform is selected. The performance is considered to be much better when compared to the gradient method and classical HE. Then classified using MD classifier, R-SM using LOOCV to recognize the faces, and performance metrics like Accuracy (Acc), Precision (Prec) and Recall (Recl). They are explained in detail. The comparison phase consists of same blocks of the classification phase but the only difference is the PCA feature extraction using three ways such as mean, median and mode. Then performance metrics like recognition accuracy (Acc) percentage and recognition time are compared.

Pre-processing using enhancement techniques

Image enhancement by intensity transformation with the histogram equalization (HE) will reduce this problem. In faces the HE is applied [4] in light background; which will leads to dark face regions face detection failure. In this proposed method, LS using GSNO and CWSNO algorithm estimate the image quality base and intensity histogram and select best transform. It performs much better than classical histogram equalization (HE) and gradient method. Transform represents the pixel values in some other equivalent form through point/neighbourhood processing [24]. The Fig. 3 shows that the neighbourhood processing operation. Any size rectangle and any shape filter are possible. Some of image processing operations involved in processing an image in sections called *blocks* or *neighbourhoods*, instead of processing the whole image at once. This approach is utilized by several functions in the tool box, such as linear filtering and morphological functions. It has two ways such as CWSNO and GSNO. Column-Wise Sliding Neighbourhood Operations (CWSNO) which provides a way of speeding up neighbourhood or block operations by rearranging blocks into matrix columns, performing this that the execution time of the image processing will be much reduced by column-wise distinct block operations and sliding neighbourhood. Single call to the mean function performs the computation faster than the individual usage of calling mean for each block. CWSNO function should be used for column processing. The function can be defined as follows:

1. Each distinct block or sliding of an image matrix is redesigned into a temporary matrix along column-wise.
2. Temporary matrix is passed to a function specified.
3. Resulting matrix is rearranged to the original shape.

Fig. 4 CWSNO creates a temporary matrix for sliding neighbourhood



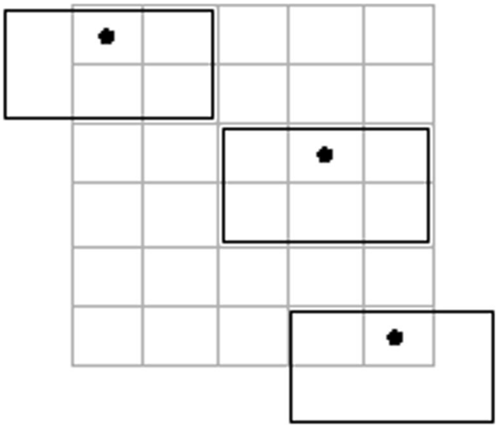


Fig. 5 Neighborhood GSNO blocks in a 6-by-5 matrix

In the sliding neighbourhood operation, the temporary matrix is created in column for each pixel with CWSNO and the corresponding pixel neighbourhood has the value of given pixel [25] and it is illustrated in Fig. 4. Zero padding of the input image is done by CWSNO which is essential for the neighbourhood of the upper left pixel. Two zero-valued neighbours are observed in the figure due to zero padding.

The temporary matrix is passed to a function, which must return a single value for each column with General Sliding Neighborhood Operation sets each output pixel to the maximum value in the input [26] pixel's neighborhood, to produce same result as the GSNO. To operate sliding neighborhood operation uses disting block processing function in an image a block at a time rather than a pixel at a time and it is shown in Fig. 5 for some of the elements in a 6-by-5 matrix with 2-by-3 GSNO sliding blocks [10, 11, 28]. The center pixel for each neighborhood is marked with a dot.

For example, in a 2-by-2 neighbourhood the center pixel is the upper left one. For any m-by-n neighbourhood, the center pixel is floor (([m n] + 1)/2). In the 2-by-3 block shown in the preceding figure, the center pixel is (1, 2) or the pixel in the second column of the top row of the neighbourhood. The sum values of the neighbourhood pixel are averaged [29] and divides by number of pixels in the neighbourhood to generate the resulting value.

Features extraction

Dimensionality has input space and data. Coordinates system suited to a set of dimensionality reduction has reduction error [20]. Principal Components Analysis (PCA) is a statistical method. Input data set has vectors/points. Input space is a space orthonormal basis. Each vector from the input space can be expressed as a linear combination of basis vector defines a dimension. Dimensions are sorted in each basis vector assigned with a weight [8]. Main directions of variance in the input set [27]. The basic principle for PCA transforms, a set of [13] correlated variables into a smaller set of uncorrelated variables called principal components. The Objective is discovering the "true dimension" of the data. It may be that p dimensional data can be represented in $q < p$ dimensions without losing much information. PCA find a lower dimensional space as shown in the Fig. 6 that best represents the data in a least-squares sense. PCA is used for reducing the dimension vector to the improvement in recognizing image [21] and found application in fields such as face recognition and image compression [22].

The following explains the PCA algorithm [14, 15] used in the classification phase:

1. Normalization of the mean: Mean dimensional values is subtracted from each value in the data by which the average of each dimension is zero.
2. Covariance matrix is calculated.
3. Eigen values and Eigen vectors are calculated.

Fig. 6 PCA in lower dimensional space

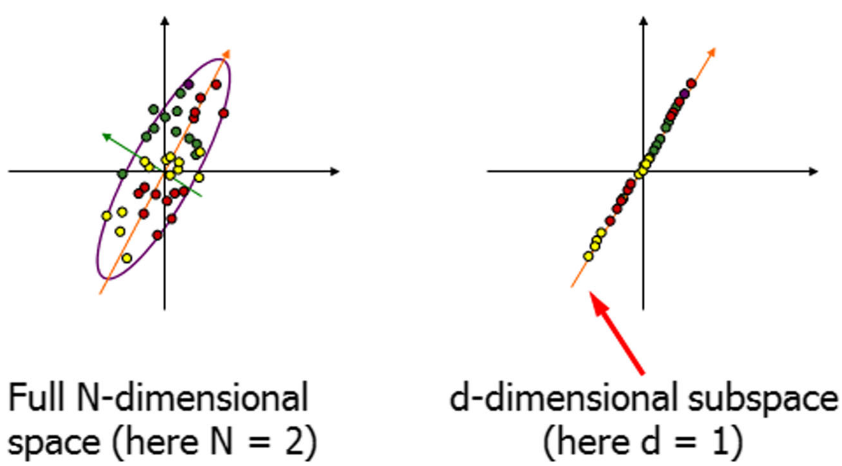


Fig. 7 Divide data sets

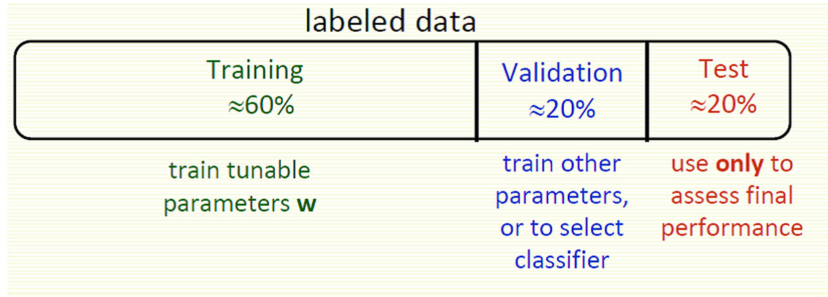


Fig. 8 CV step

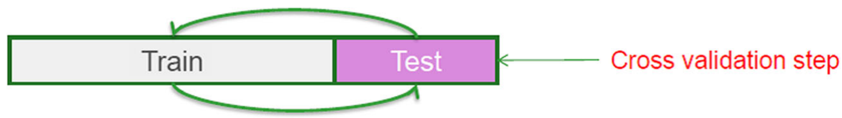


Fig. 9 Predict step

4. Eigen vector is nothing but each new axis of the data and Eigen value is the standard deviation of the data variance of each new axis for that Eigen vector.
5. Ranking and normalization of Eigen vectors by Eigen values is to be done.
6. Feature vector is to be formed by stacking top k Eigen vectors.
7. Data transformation to PCs:

   New data = feature vectors (transposed) * Original data as shown in the Eq. (1).

Several industries, forensic disciplines and medical oriented applications rely on this idea.

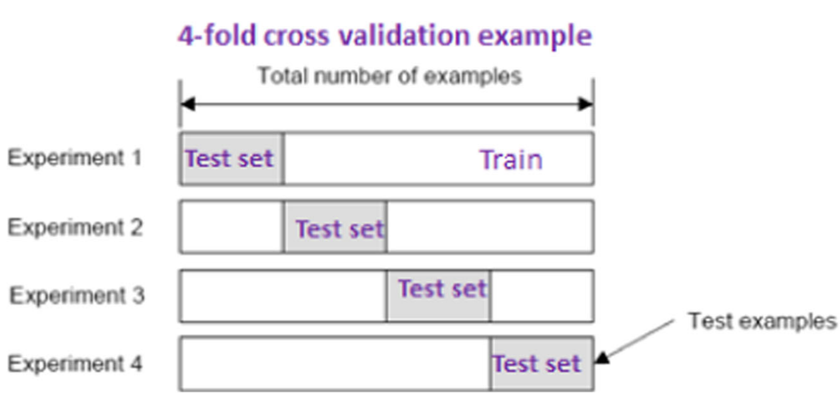
$$\begin{pmatrix} y1 \\ \vdots \\ yK \end{pmatrix} = \begin{pmatrix} u11 & \cdots & uK1 \\ \vdots & \ddots & \vdots \\ u1n & \cdots & uKn \end{pmatrix}^T \begin{pmatrix} x1 \\ \vdots \\ xn \end{pmatrix} \tag{1}$$

In the comparison phase of the PCA algorithm, the same steps is followed for classification phase, the only difference is the calculation for normalization of median and mode values which is done additionally.

Set classifiers

In the test set is examples used only to assess the performance of fully-trained classifier. After assessing the model with the test set, YOU MUST NOT further tunes your model (that's the theory anyway in order to prevent 'learning the test set' and 'overfitting'). In our proposed method, we used [9, 15] Minimum Distance (MD) Classifier. Euclidean Distance is most commonly used to measure distance. The Euclidean Distance is probably the most widely used distance metric. It is a special case of a general class of norms and is given as in the Eq. (2).

Fig. 10 4-fold CV example.



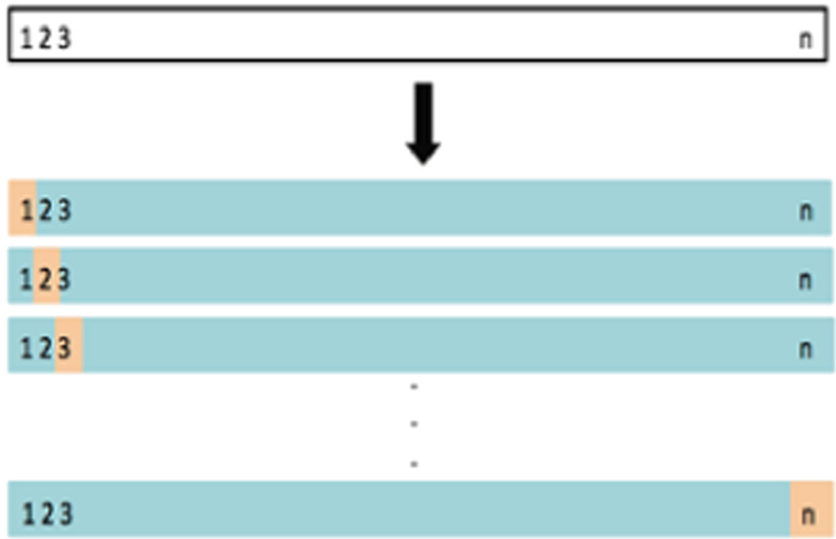


Fig. 11 LOOCV operation

$$d_E(x,y) = \sqrt{(x_1-y_1)^2 + (x_2-y_2)^2 + \ldots + (x_n-y_n)^2}$$
$$= \sqrt{\sum_{i=1}^{n} (x_i-y_i)^2} \qquad (2)$$

The Euclidean Distance takes into account both the direction and the magnitude of the vectors. The Euclidean Distance between two n-dimensional vectors $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots y_n)$. Each axis represents [18] an experimental sample. The co-ordinate on each axis is the measure of expression level of a gene in this sample.

Re-sampling method

The Re-Sampling Method (R-SM) involves repeatedly drawing samples from a training set and refitting a model of interest on each to estimate the variability of a linear regression. It is done by repeatedly drawing different samples from the training data, fitting a [19] regression to each new sample and then examining the extent to which the resulting fits differ. In the model assessment, a final model is chosen, estimating its prediction error on new data.

But these days, we have advanced equipments. Two types of R-SM are cross validation and boot strapping. Cross-Validation (CV) used to estimate test set prediction error rates associated with a given machine learning method to evaluate its performance or to select the appropriate level of model flexibility. Boot strap is used most commonly to provide a measure of accuracy of a parameter estimate or of a given machine learning method, which is proposed in this paper. This process is sometimes referred to as “validating” the regression equation. One way to address this issue is to obtain a new sample of observations. In cross-validation the original sample is split into two parts. One part is called the training (or

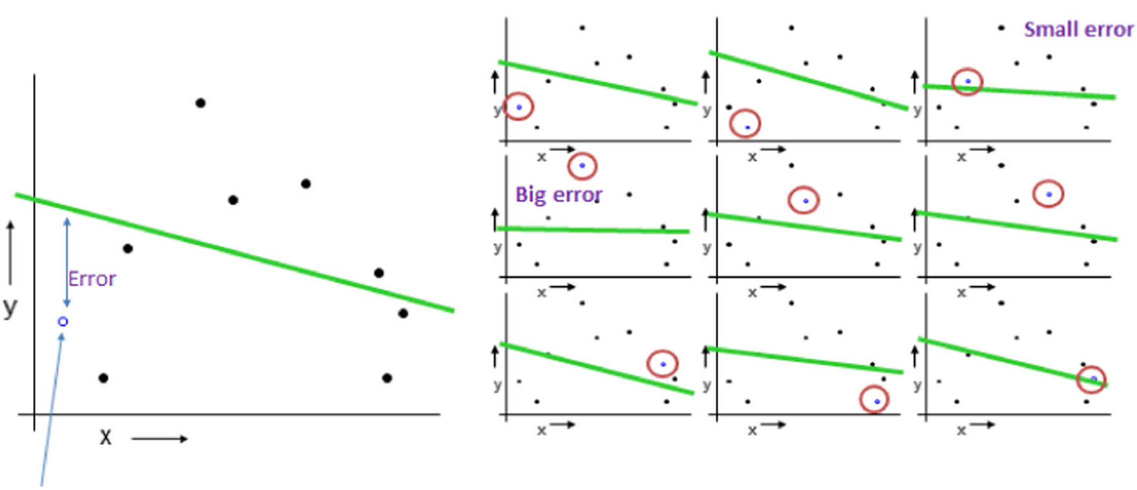


Fig. 12 LOOCV operation

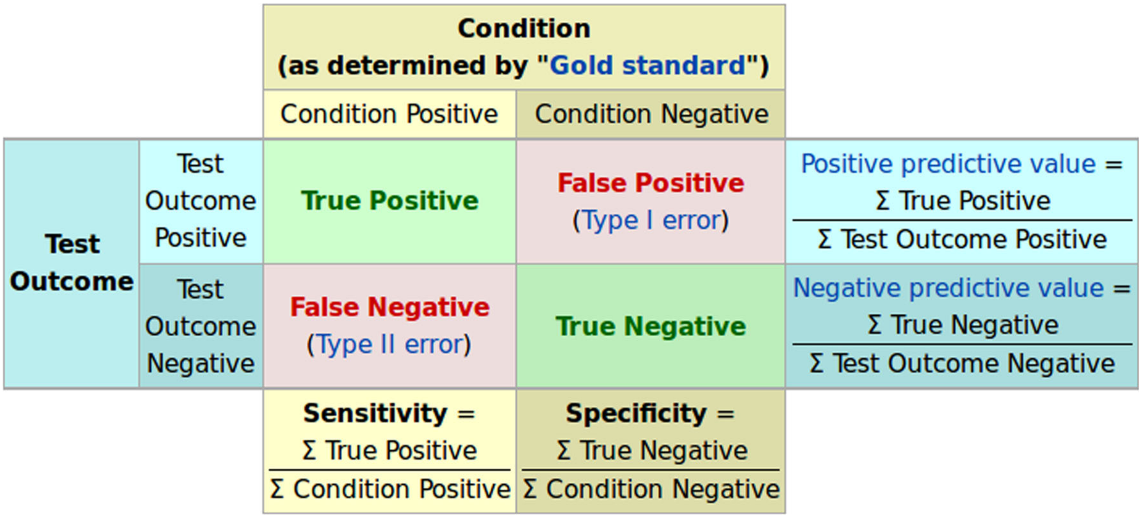


Fig. 13 Confusion matrix

derivation) sample, and the other part is called the validation (or validation + testing) sample. If sample size is very large, it is often best to split the sample in half. For smaller samples, it is more conventional to split the sample such that 2/3 of the observations are in the derivation sample and 1/3 is in the validation sample. The most common approach is to divide the sample randomly, thus theoretically eliminating any systematic differences. Another approach is to define matched pairs of subjects in the original sample and to assign one member of each pair to the derivation sample and the other to the validation sample.

The data was divided into three sets:- training, validation and test sets, which are shown in the Fig. 7. The optimal model on the training set, and use the test set to check CV

Fig. 14 Sample database images

Fig. 15 Pre-processing output

its predictive capability as shown in the Fig. 8. The model can predict the test set as shown in the Fig. 9. The validation error gives an unbiased estimate of the predictive power of a model.

In CV model, this proposed method used LOOCV (Leave One Out Cross Validation). It is used when a hold-out plan is difficult to design or was not designed at the beginning the data set is small and all the data should be used for good model training. Create a K-fold partition of the dataset. For each of the K experiments use K-1 folds for training and the remaining one for testing. The Fig. 10 shows that the 4-fold CV example.

In LOOCV, it is an average usually average of performance between experiments as in the Eq. (3).

$$E = 1/K\sum_{i=1}^{k} E_i \tag{3}$$

Instead of creating two subsets of comparable size, a single observation is used for the validation set and the remaining observations (n – 1) make up the training set. LOOCV operation is the degenerate case of K-Fold cross validation, where K = n is chosen as the total number of examples as shown in the Fig. 11.

LOOCV algorithm is as follows:

i. Split the entire data set of size n into:
   1. Training data set.
   2. Validation data set.

ii. Fit the model using the training data set.

iii. Evaluate the model using validation set and compute the corresponding MSE.

iv. Repeat this process n times, producing n squared errors. The average of these n squared errors estimates the test MSE (Mean Square Error) as in the Eq. (4).

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n} MSE_i \tag{4}$$

For example, k = 1 to R. Let (x_k,y_k) be the k^{th} record. Temporarily remove (x_k,y_k) from the dataset. Train on the remaining R-1 datapoints. Note your error (x_k,y_k). After you've done all points, report the mean error. The Fig. 12 shows that the LOOCV operation.

LOOCV has far less bias and therefore, tends not to [21] overestimate the test error rate. Performing LOOCV multiple times always yields the same results because there is no randomness in the training/validation set splits. LOOCV is computationally intensive because the model has to be fit n times. However, there is a shortcut with linear or polynomial regression (where h_i is the leverage) as in the Eq. (5).

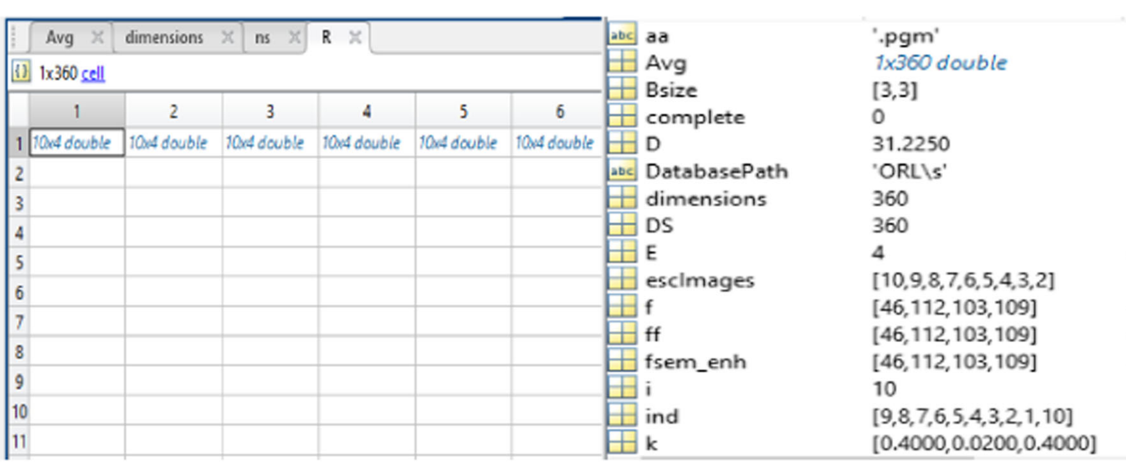


Fig. 16 PCA features data matrix result

$$\mathrm{CV}_{(\mathrm{n})}=\frac{1}{\mathrm{n}}\sum_{\mathrm{i}=1}^{\mathrm{n}}\left(\frac{\mathrm{Y}_{i}-\hat{\mathrm{Y}}_{\mathrm{i}}}{1-\mathrm{h}_{\mathrm{i}}}\right)^{2} \tag{5}$$

The validation approach produces different MSE when applied repeatedly due to randomness in the splitting process, while performing LOOCV multiple [22] times will always yield the same results, because we split based on 1 obs. each time. LOOCV is computationally intensive. Let's, fit the each model n times.

Performance metrics

By classifying the patients with respect to the test results and the true disease status, the following confusion matrix is obtained as is shown in Fig. 13. Many performance terms are precision, recall, sensitivity, specificity, true negative rate, true positive rate, PPV, NPV, type 1 error and type 2 error as is shown in Fig. 13.

Accuracy (Acc), Positive Predictive Value (PPV) or Precision (Prec) and Negative Predictive Value (NPV) or Recall (Recl) is obtained. Three performance assessments are plotted graph in classification phase. The ideal value of the PPV, NPV with a perfect test is 1 and the worst possible value would be zero. The recognition Accuracy (Acc) percentage and time are calculated and compared with three PCA algorithm techniques like mean, median and mode of comparison phase [7]. The performance assessment of proposed methodology is analyzed with the following parameters using this formula from Eqs. (6) to (8).

$$\frac{Precision}{Positive\ Predictive\ Value\ (PPV)}=\frac{TP}{TP+FP}=\frac{TP}{nP} \tag{6}$$

$$\frac{Recall}{Negative\ Predictive\ Value\ (NPV)}=\frac{TN}{TN+FN}=\frac{TN}{nN} \tag{7}$$

$$Accuracy\ (Acc)=\frac{(TP+TN)}{(TP+FN+TN+FP)} \tag{8}$$

Results and discussion

The algorithm has been implemented in MATLAB code, in version R2014. The Fig. 14 shows the sample database images. These images are collected from web resource. Origin of ORL (Olivetti Research Laboratory), was found in

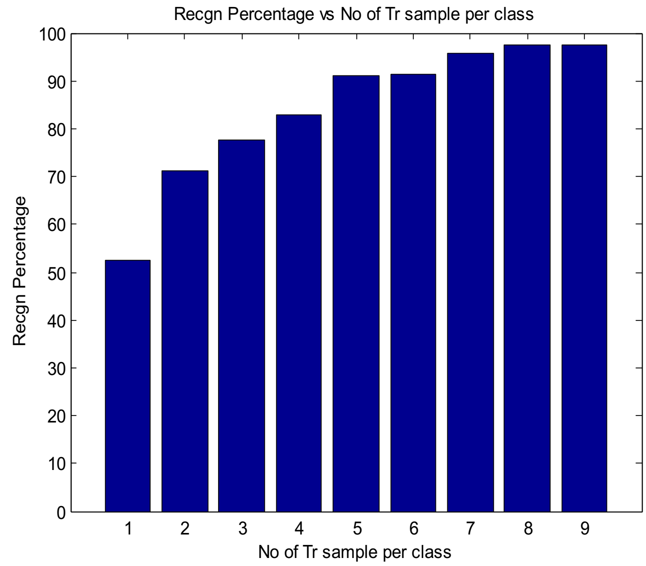


Fig. 17 Accuracy (Acc) performance on MD classifier with LOOCV of classification phase using LS-GSNO

1992~1994. The statistics ORL database has 40 subjects, each with 10 images. Characteristics are same-size photos and centered faces of different pose which suits well for evaluation.

Pre-processing output

The input image is taken in the Portable Grey Map (PGM) file format [23]. For pre-processing, first it is applied for LS-CWSNO and GSNO enhancement techniques. The Fig. 15 shows that the pre-processing output.

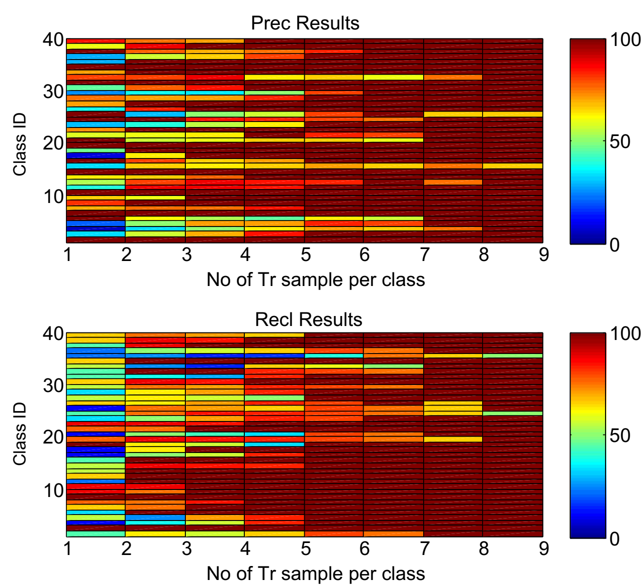


Fig. 18 Precision (Prec) and Recall (Recl) performance on MD classifier with LOOCV of classification phase using LS-GSNO

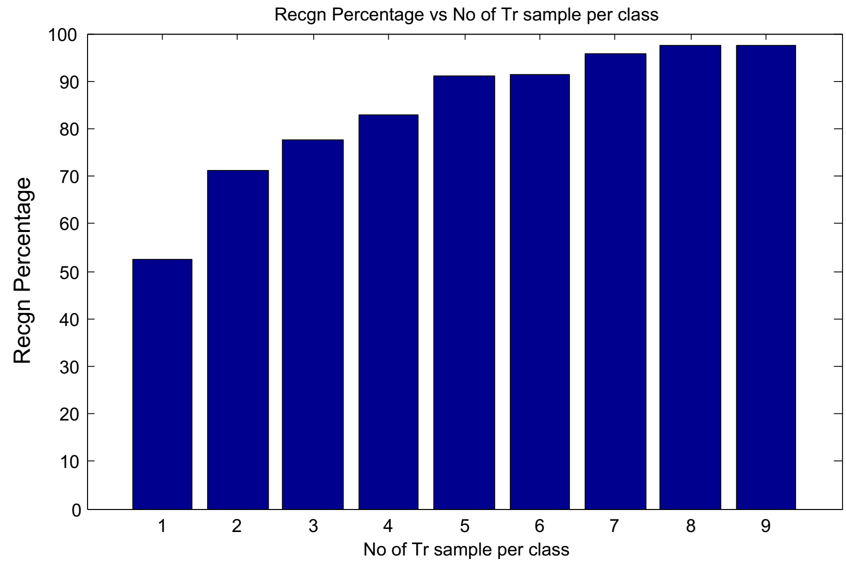


Fig. 19 Accuracy (Acc) performance on MD classifier with LOOCV of classification phase using LS-CWSNO

Features extraction result

In this section, PCA features data matrix result of pre-processed images is extracted. The Fig. 16 shows that the PCA features data matrix result in the MATLAB [6] variable and workspace window.

Classification phase and its performance outputs

In this section, the PCA performance assessments like Accuracy (Acc)/ recognition percentage, Positive Predictive Value (PPV)/ Precision (Prec) and Negative Predictive Value (NPV)/ Recall (Recl) using MD classifier with LOOCV are calculated. The Fig. 17 shows the Accuracy (Acc) performance on MD classifier with LOOCV of classification phase using LS-GSNO. The Fig. 18 shows the Precision (Prec) and Recall (Recl) performance on MD classifier with LOOCV of classification phase using LS-GSNO. The Fig. 19 shows the Accuracy (Acc) performance on MD classifier with LOOCV of classification phase using LS-CWSNO. The Fig. 20 shows the Precision (Prec) and Recall (Recl) performance on MD classifier with LOOCV of classification phase using LS-CWSNO.

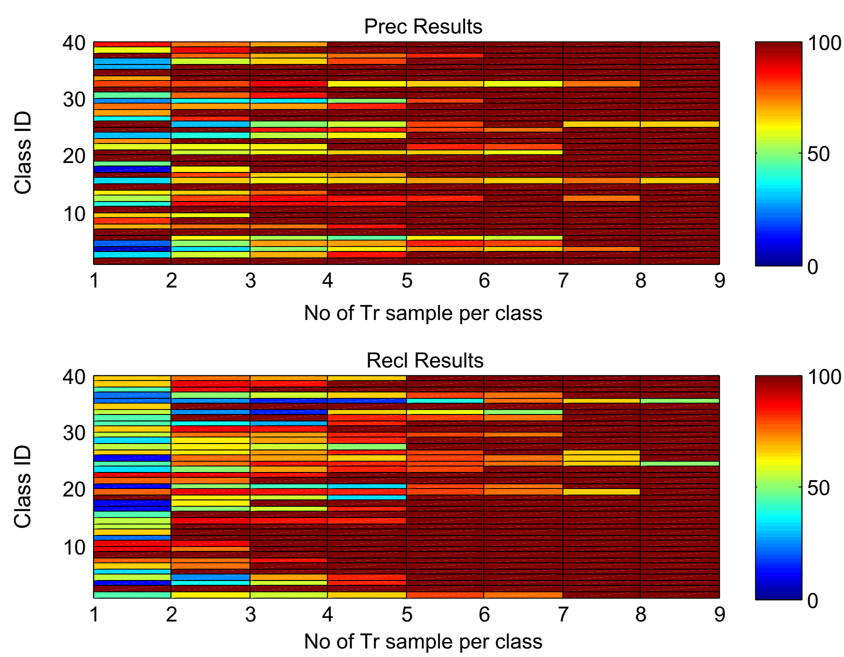


Fig. 20 Precision (Prec) and Recall (Recl) performance on MD classifier with LOOCV of classification phase using LS-CWSNO

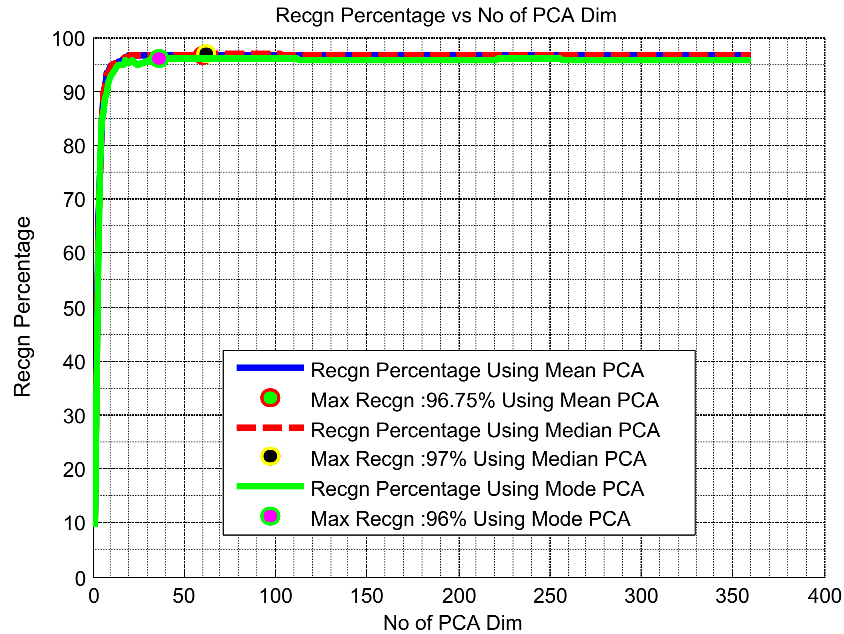


Fig. 21 Recognition percentage performance using mean, median and mode PCA features on MD classifier with LOOCV of comparison phase using LS-GSNO

Comparison phase and its performance outputs

In this section, calculate the PCA performance assessments like Accuracy (Acc)/ recognition percentage and recognition timeversus number of PCA dimensions using MD classifier with LOOCV.

The Fig. 21 shows that the recognition percentage performance using mean (96.75%), median (97%) and mode (96%) PCA features on MD classifier with LOOCV of comparison phase using LS-GSNO indicates better performance. The Fig. 22 shows the recognition time performance using mean, median and mode PCA features on MD classifier with LOOCV of comparison phase using LS-GSNO. The Fig. 23 shows the recognition percentage performance using mean, median and mode PCA features on MD classifier with LOOCV of comparison phase using

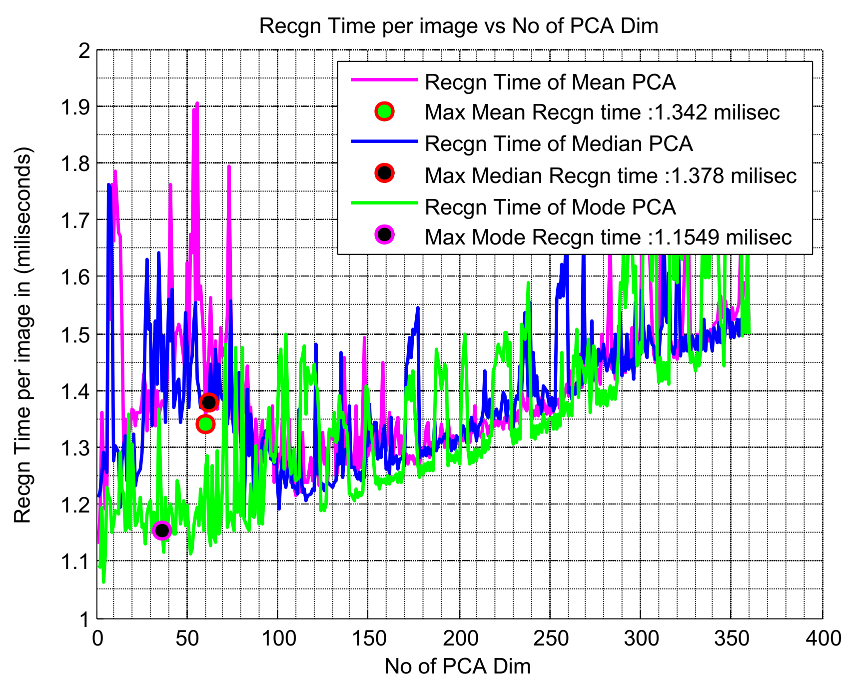


Fig. 22 Recognition time performance using mean, median and mode PCA features on MD classifier with LOOCV of comparison phase using LS-GSNO

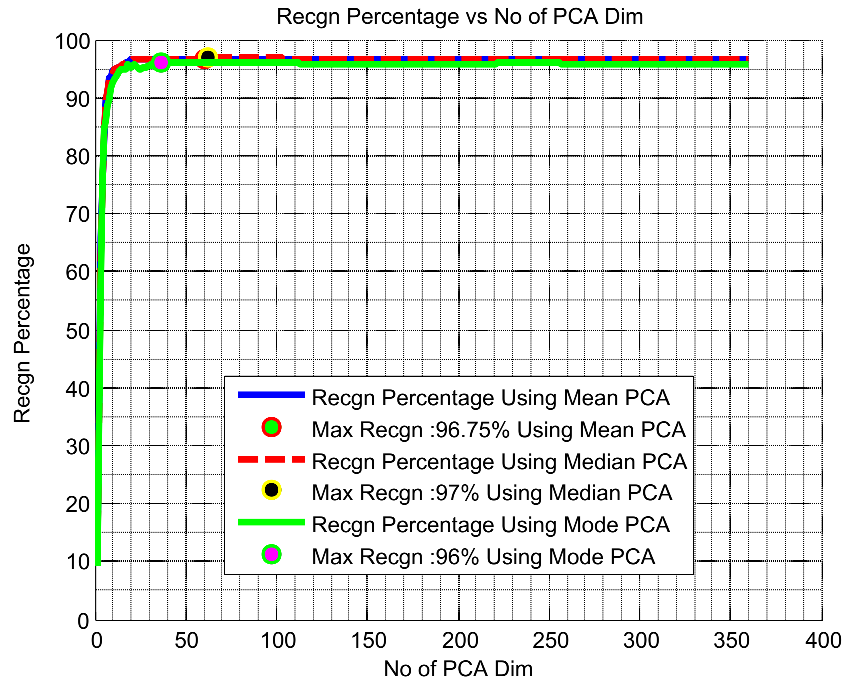


Fig. 23 Recognition percentage performance using mean, median and mode PCA features on MD classifier with LOOCV of comparison phase using LS-CWSNO

LS-CWSNO. The Fig.newnbsp;24 shows the recognition time performance using mean, median and mode PCA features on MD classifier with LOOCV of comparison phase using LS-CWSNO. The mean PCA recognition time is 4.7184 millisec, median PCA recognition time is 2.0011 millisec and mode PCA recognition time is 1.1181 millisec. The mean PCA recognition time duration is high compared to mode PCA recognition time.

Conclusion

In this proposed method, a face recognition system has two phases like classification and comparison with LS enhancement technique using GSNO and CWSNO- Then, the PCA features are extracted with three ways such as mean, median and mode, and are then classified with MD classifier using LOOCV of R-SM to recognize the faces. The Accuracy (Acc),

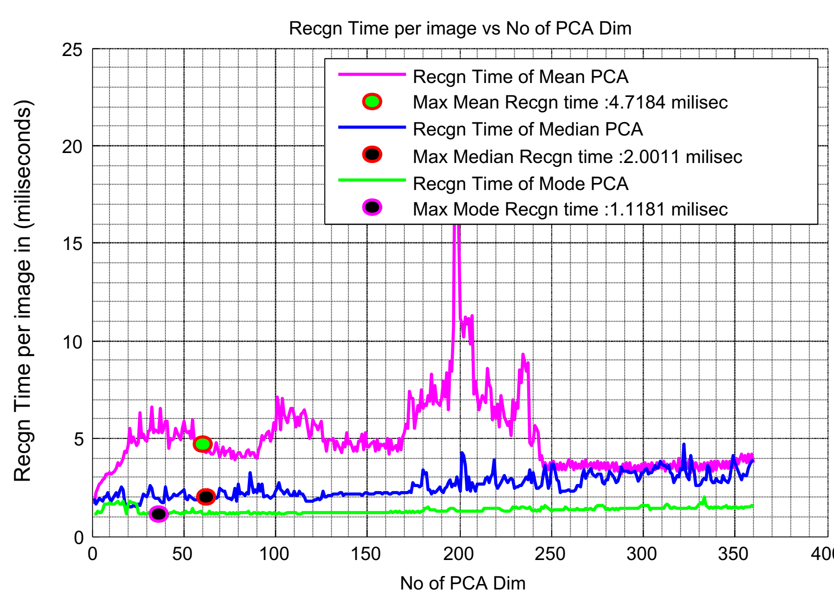


Fig. 24 Recognition time performance using mean, median and mode PCA features on MD classifier with LOOCV of comparison phase using LS-CWSNO

Precision (Prec) and Recall (Recl) performance of MD classifier has been analysed and compared with two different enhancement processes like LS-GSNO and LS-CWSNO of PCA features. From that result, the classifier gives better classification rate when compared with two enhancement processes (LS-GSNO: 99% and LS-CWSNO: 98% respectively). Finally, performance assessments of Precision (Prec) and Recall (Recl) are plotted by two ways of enhancement approaches. In addition, PCA features of mean, median and mode for two enhancement processes are obtained. Evaluation results of recognition percentage and time by using MD with LOOCV of R-SM approach based on two enhancement processes (Recognition Percentage/ Accuracy (Acc) for mean, median and mode PCA features are 96.75, 97 and 96% respectively) indicates better performance. The recognition time for mean, median and mode PCA features for LS-GSNO are 1.342 millisec, 1.378 millisec and 1.1549 millisec respectively. The recognition time for mean, median and mode PCA features for LS-CWSNO are 4.7184 millisec, 2.0011 millisec and 1.1181 millisec respectively. The implementation result is taken on the ORL face database. This idea is mainly applied in several medical oriented applications, industries and forensic disciplines.

Compliance with Ethical Standards

Conflict of Interest All authors declares that they have no conflict of interest in publishing this article.

References

1. Chellappa, R., Wilson, C. I., and Sirohey, S., Human and machine recognition of faces: A survey. *Proc. IEEE* 83(5):705–741, 1995.
2. Zhao, W., Chellappa, R., Rosenfeld, A., and Phillips, P. J., Face recognition: A literature survey. *ACM Comput. Surv.* 35(4):399–458, 2003.
3. Ramamoorthi, R., and Hanrahan, P., On the relationship between radiance and irradiance: Determining the illumination from images of a convex Lambertian object. *J. Opt. Soc. Amer.* 18(10):2448–2459, 2001.
4. Gross, R., and Brajovie, V., An image preprocessing algorithm for illumination invariant face recognition. In: *Proc. 4th Int. Conf. Audio Video Based Biometric Person Authentication*, Vol. 2688, 2003, 10–18.
5. Fan, X., and Tjahjadi, T., A dynamic frame work based on local Zernike moment and motion history image for facial expression recognition. *Pattern Recogn.* 64:399–406, 2017.
6. Gonzalez, R. C., and Woods, *Digital image processing using MATLAB*. New Delhi: Pearson prentice Hall, ch. 6, 2000, 424–428.
7. Nhat, V. D. M., and Lee, S., An improvement on PCA algorithm for face recognition. In: *Advances in Neural Networks - ISNN 2005*. Vol. 3498, Lecture notes in computer science. Chongqing: Springer, 2005, 1016–1021.
8. Hwang, H., and Haddad, R. A., Adaptive median filters: New algorithms and results. *IEEE Trans. Image Process.* 4(4):499–502, 1995.
9. Russ, J. C., *The image processing handbook*. Second edition. Boca Raton: CRC Press, 1995.
10. Castleman, K. R., *Digital image processing*. Second edition. Englewood Cliffs: Prentice Hall, 1996.
11. James, G., Witten, D., Hastie, T., and Tibshirani, R., *An introduction to statistical learning, with applications in R*. New York: Springer, 2013.
12. Liang, Y., Gong, W., Pan, Y., Li, W., and Hu, Z., Gabor features-based classification using SVM for face recognition. In: *Advances in Neural Networks – ISNN 2005*. Vol. 3497, Lecture notes in computer science. Chongqing: Springer, 2005, 118–123.
13. Chen, S., and Zhu, Y., Subpattern-based principle component analyses. *Pattern Recogn.* 37(1):1081–1083, 2004.
14. Gottumukkal, R., and Asari, V. K., An improved face recognition technique based on modular PCA approach. *Pattern Recogn. Lett.* 25(4):429–436, 2004.
15. Tan, K., and Chen, S., Adaptively weighted sub-pattern PCA for face recognition. *Neuro Comp.* 64:505–511, 2005.
16. Abu-Mostafa, Y., Magdon-Ismail, M., and Lin, H., Learning from data: a short course. AML Book Publisher: USA, 2012.
17. Turk, M., and Pentland, A., Face recognition using eigenfaces. Conference on computer vision and pattern recognition, 3–6 June 1991, Maui, HI, USA, 586–591.
18. Daugman, J. G., High confidence visual recognition of persons by a test of statistical independence. *IEEE Trans. Pattern Anal. Mach. Intell.* 15(11):1148–1160, 1993.
19. Jain, A. K., Ross, A., and Prabhakar, S., An introduction to biometric recognition. *IEEE Transaction Circuits System Video Technologies* 14(1):4–20, 2004.
20. Singh, C., Walia, E., and Mittal, N., Rotation invariant complex Zernike moments features and their applications to human face and character recognition. *IET Comput. Vis.* 5(5):255–265, 2011.
21. Wang, Z., and Li, X., Face recognition based on improved PCA reconstruction. In: *2010 8th world congress on Intelligent Control and Automation (WCICA)*, 2010, 6272–6276.
22. Meng, J., and Yang, Y., Symmetrical two-dimensional PCA with image measures in face recognition. *Int. J. Adv. Robot. Syst.* 9(6):238, 2012.
23. Poskanzer, J., PGM file extension. Wikipedia.
24. Li, S. Z., and Jain, A. K., *Handbook of face recognition*. 2005
25. Bichsel, M., and Pentland, A. P., Human face recognition sand the face image set's topology. *CVGIP. Image Understanding* 59(2):254–261, 1994.
26. Fukunaga, K., *Introduction to statistical pattern recognition*. Boston: Academic Press, 1990.
27. Shapiro, L. G., and Stockman, G. C., *Computer vision*. Upper Saddle River: Prentence Hall, 2001, 137–150.
28. Peng, S., and Lucke, L., A hybrid filter for image enhancement. *International Conference on Image Processing* 1:163–166, 1995.
29. Sridhar, S., *Digital image processing*. New Delhi: Oxford University press, ch. 5, 2013, 193–195.