

PCA-based dimensionality reduction for face recognition

Md. Abu Marjan¹, Md. Rashedul Islam², Md. Palash Uddin³, Masud Ibn Afjal⁴, Md. Al Mamun⁵

^{1,3,4}Department of Computer Science and Engineering, Hajee Mohammad Danesh Science and Technology University, Bangladesh

²Information Technology Cell, Hajee Mohammad Danesh Science and Technology University, Bangladesh

⁵Department of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Bangladesh

Article Info

Article history:

Received Jan 6, 2021

Revised Mar 19, 2021

Accepted Mar 30, 2021

Keywords:

Data reduction

Dimensionality reduction

Eigen analysis

Face recognition

Principal component analysis

ABSTRACT

In this paper, we conduct a comprehensive study on dimensionality reduction (DR) techniques and discuss the mostly used statistical DR technique called principal component analysis (PCA) in detail with a view to addressing the classical face recognition problem. Therefore, we, more devotedly, propose a solution to either a typical face or individual face recognition based on the principal components, which are constructed using PCA on the face images. We simulate the proposed solution with several training and test sets of manually captured face images and also with the popular Olivetti Research Laboratory (ORL) and Yale face databases. The performance measure of the proposed face recognizer signifies its superiority.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Md. Abu Marjan

Department of Computer Science and Engineering

Hajee Mohammad Danesh Science and Technology University

Dinajpur-5200, Bangladesh

Email: marjan@hstu.ac.bd

1. INTRODUCTION

Data mining is a way for extracting or mining knowledge from large amounts of data [1]-[4]. In developing data mining application, the amount data taken from various repositories such as databases, data warehouse, and World Wide Web (WWW). is typically huge to be either stored or processed. Long time may be required for analyzing complex data and mining on huge amounts of data. Therefore, it makes such analysis sometimes impractical or infeasible. Data reduction techniques are traditionally applied to find a reduced representation of the dataset, which is much smaller in size ensuring the close integrity of the original data. To what follows, mining on the reduced dataset should be more efficient producing the same or almost the same analytical results. The common strategies for data reduction include data cube aggregation, attribute subset selection, dimensionality reduction (DR) and numerosity reduction [1].

Recently, the dataset size in terms of number of records and attributes is exploring very rapidly, which prompts the development of a number of big-data platforms, parallel data analytics algorithms and the usage of data DR procedures efficiently. In order to handle the real-world data effectively, the respective dimensionality needs to be reduced in an effective (more economic) amount. DR is the study of methods of transformations for reducing the number of dimensions describing the object of high-dimensional data into a meaningful representation of reduced dimensionality. Theoretically, the reduced representation of dataset should have such a dimensionality that corresponds to the intrinsic dimensionality of the dataset. The intrinsic dimensionality of

dataset means the minimum number of arguments needed to account for the observed properties of the data. The general objectives of DR are to remove irrelevant and redundant data for reducing the manipulation cost and avoiding data over-fitting, and increasing the quality of data for efficient data-intensive processing tasks, such as pattern recognition, data mining, visualization, database navigation, and compression of high-dimensional data. As such, DR offers an effective solution to the diverse problem of “curse of dimensionality” and fixes other undesired properties of high-dimensional spaces [5].

Mathematically, the DR techniques can be defined as to convert a given dataset represented in a $n \times D$ matrix X consisting of n data vectors $x_i, i = 1, 2, \dots, n$ with dimensionality D into another dataset Y that has an intrinsic dimensionality d , where $d < D$, and often $d \ll D$. The intrinsic dimensionality of data signifies that the points in dataset X are belonging to or near a manifold with dimensionality d that is implanted in the D dimensional space. In another words, the DR methods encode the given dataset X having dimensionality D into a new dataset Y with dimensionality d retaining the geometry of the data as much as possible. In general, neither the intrinsic dimensionality d of the dataset X nor the geometry of the data manifold is completely known. Therefore, DR of a dataset is an ill-posed problem that can only be solved by assuming certain properties of the data such as its intrinsic dimensionality [5].

There are some DR techniques for the purpose of taking a smaller image and compression and there are some other DR techniques for machine learning purpose (e.g., for better data analysis, classification, statistics, and visualization) [6]. In machine learning, dimension reduction is usually concerning with the feature vectors. In this case, DR techniques can be divided into two categories: feature extraction and feature selection methods. Feature extraction can further be divided into linear and non-linear methods. The main goal of some methods is to preserve fidelity with respect to the original data using a certain metric such as mean squared error, and the goal of some other methods is to improve the performance of a typical task, such as classification, prediction, and visualization [7]. Linear feature extraction methods include principal component analysis (PCA), factor analysis, independent component analysis (ICA), and linear discriminant analysis (LDA). Non-linear feature extraction methods include the front-ranked techniques such as multidimensional scaling (MDS), Isomap, maximum variance unfolding, kernel PCA etc [5]. Feature selection is divided into feature ranking and feature subset selection. Feature ranking commonly uses two scoring function, such as Euclidean distance and correlation and information gain ratio. On the other hand, the feature subset selection methods are divided into filter method, wrapper method and embedded method. The filter methods do not use any learning algorithm [8].

In this paper, after conducting a comprehensive study on the DR techniques, we present a face recognition approach using PCA transformation. We perform experiment using Olivetti Research Laboratory (ORL) and Yale face databases. The experimental results manifest the superiority of the proposed method. The main contribution of this paper is listed: i) comprehensive study on the DR techniques; ii) technical and mathematical intuitions behind the PCA approach; iii) two face recognition proposals using PCA data; and iv) performance evaluation on ORL and Yale face databases.

The remainder of this paper is organized as follows. We provide the technical detail of the PCA method in section 2. Then, we discuss the related works to ours in section 3. After that, we explain the proposed face recognition approach in section 4. The experiments and results are provided in section 5. At last, we summarize and conclude the findings and observations in section 6.

2. PRINCIPAL COMPONENT ANALYSIS

The constituent attributes of real-world dataset reveal relationships among them. The relationships are often linear or approximately linear. This makes the attributes amenable to common analysis techniques. One of such techniques is PCA, which rotates the original data to new coordinates with a view to making the data as flat as possible. PCA is a statistical transformation that identifies patterns in data through detecting the correlation between attributes [9]. If there exists a strong correlation between attributes, the attempt to reduce the dimensionality only makes sense. PCA finds the directions of maximum variance in high-dimensional data and then projects it onto a reduced dimensional subspace while retaining most of the information of the original dataset [10]. Mathematically, given a matrix of two or more attributes, PCA produces a new matrix with the same number of attributes, called the principal components. Each generated principal component is a linear transformation of the entire original dataset. The measurements of the principal components are calculated in such a way that the first principal component holds the maximum variance, which can tentatively

be thought as the maximum information. The second principal component is calculated to have the second most variance, and, significantly, in a linear sense is uncorrelated with the first principal component. The further principal components, if there are any, exhibit decreasing variance and are uncorrelated with all other principal components. The steps for the implementation of PCA are illustrated [11]:

- Step 1: Take the whole dataset consisting of d -dimensional samples ignoring the class labels.
- Step 2: Compute the d -dimensional mean vector. The mean vector consists of the means of each variable. The mean is the sum of the data points divided by the number of data points. That is, $\mu = \bar{\mathbf{A}} = \sum_{i=1}^n \frac{A_i}{n}$. The mean is that value that is most commonly referred to as the average. The mean vector is often referred to as the centroid. The variance is roughly the arithmetic average of the squared distance from the mean. The variance is defined as $\sigma^2 = s^2 = \text{var}(\mathbf{A}) = \sum_{i=1}^n \frac{(A_i - \bar{\mathbf{A}})^2}{n-1}$, where $\bar{\mathbf{A}}$ is the mean of the data. Note that the standard deviation (σ) is the square root of the variance.
- Step 3: Compute the covariance matrix, alternatively, the scatter matrix of the whole dataset.

- a. Covariance matrix: The variance-covariance matrix consists of the variances of the variables along the main diagonal and the covariances between each pair of variables in the other matrix positions. The formula for computing the covariance of the variables S and T is $\text{covar}(S, T) = \frac{\sum_{i=1}^n (S_i - \bar{S})(T_i - \bar{T})}{n-1}$, where \bar{S} and \bar{T} denote the means of S and T , respectively. The covariance matrix is defined as

$$\Sigma = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \dots & \sigma_{1n}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \dots & \sigma_{2n}^2 \\ \dots & \dots & \dots & \dots \\ \sigma_{n1}^2 & \sigma_{n2}^2 & \dots & \sigma_{nn}^2 \end{bmatrix}$$

Here, σ_{ii}^2 is the variance of each variable A_i in \mathbf{A} , σ_{jk}^2 is the covariance between A_j and A_k in \mathbf{A} .

- b. Scatter matrix: The scatter matrix is computed as $\sum_{i=1}^n (A_i - m)(A_i - m)^\tau$, where m is the mean vector and it is defined as $m = \sum_{i=1}^n \frac{A_i}{n}$

- Step 4: Perform eigendecomposition i.e., compute eigenvectors (e_1, e_2, \dots, e_d) and corresponding eigenvalues ($\lambda_1, \lambda_2, \dots, \lambda_d$). The eigenvectors or principal components determine the directions of the new feature space, and the eigenvalues determine their magnitude.
- Step 5: Sort the eigenvectors by decreasing eigenvalues and choose k eigenvectors with the largest eigenvalues to form a $d \times k$ dimensional matrix \mathbf{W} , where every column represents an eigenvector and k is the number of dimensions of the new feature subspace with $k \leq d$.
- Step 6: Use the $d \times k$ eigenvector projection matrix, \mathbf{W} to transform the original samples onto the new subspace. This can be summarized by the mathematical equation: $y^\tau = W^\tau \times x^\tau$, where x is a $1 \times d$ -dimensional vector representing one sample, and y is the transformed $1 \times k$ -dimensional sample in the new subspace. Alternatively, this can be performed as $\mathbf{Y} = \mathbf{A} \times \mathbf{W}$ (or $\mathbf{Y}^\tau = \mathbf{W}^\tau \mathbf{A}^\tau$), where \mathbf{Y} is the transformed $n \times k$ -dimensional samples in the new subspace.

3. RELATED WORK

Dash *et al.* [12] presented a PCA based entropy measure for ranking features and compares with a similar feature ranking method (Relief) in [12]. Maaten, Postma, and Herik have investigated the performances of the nonlinear techniques on artificial and natural tasks, also conduct review and systematic comparison of DR techniques [5]. Spectral DR methods have explained with a short tutorial in the following paper [13]. In review work [14], the authors categorized the plethora of available DR methods and illustrated the mathematical insight behind them. Looga, Ginneken, and Duin have proposed a DR technique for image features using the canonical contextual correlation projection in [15]. In [16] article, the authors provide a comprehensive review and comparison of the performance of the principal methods of dimension reduction proposed in the approximate Bayesian computation literature. Silipo, Adae, and Berthold have discussed seven techniques for DR which are missing values, low variance filter, high correlation filter, PCA, random forests, backward feature elimination, and forward feature construction in [17]. Joshi and Machchhar [18] conduct a comprehensive survey on DR methods and proposed a DR method that depends upon the given set of parameters and varying conditions [18]. The authors investigate that recursive feature elimination, and genetic and evolutionary feature weighting and selection give better classification result than PCA [19].

Several works have also been conducted on recognition problem based on PCA in various ways. Huang and Yin [20] compare and investigate linear PCA and various nonlinear techniques for face recognition. Alkandari and Aljaber [21] have presented the importance of PCA to identify the facial image without human intervention [21]. Dandpat and Meher proposed a face recognition for improving performance using PCA and two-dimensional PCA in [22]. PCA in linear discriminant analysis space for face recognition has been proposed by Su and Wang [23]. The following paper investigates the performance when two DR methods such as self-organizing map (SOM) and PCA have been combined [24].

4. PROPOSED APPROACH TO FACE RECOGNITION

In this paper, after discussing the working principle of PCA in detail, we propose a solution for face recognition problem based the principal components of the training grayscale face image matrices. The proposal is a customization of various principal components-based existing classifiers. The main customization is made in case of deriving the training and test sets, where the images are placed as matrices rather than as vectors of the traditional approaches and introducing the transpose of the main sets as discussed later. To implement the proposal, the face recognition problem is divided into two categories.

4.1. Problem statement-1: Recognition of a typical face

Given a new image, classify it to “face” or “non-face” from a set of N original peoples’ face images, each image is R pixels high by C pixels wide i.e., the pixel resolution is $R \times C$. To solve this, we merge N training image matrices into a single big matrix by placing one after another. Then, we also place the input image matrix N times one after another to form another big matrix. After that, we take the transpose of both big matrices. Subsequently, we apply PCA on the four big matrices and select k eigenvectors for each. We then determine the similarity of the normal input big matrix with the normal training big matrix, and transposed input big matrix with the transposed training big matrix using selected k features (eigenvectors). Finally, the decision is taken based the similarity result. The solution is illustrated with the following steps:

- Step 1: Input the N original images of size $R \times C$.
- Step 2: For each of the N images, convert the image to a matrix of length (dimension) $R \times C$
 - a. Step 2.1: Put all the matrices together in one big image-matrix, **Train1** like this:

$$\mathbf{Train1} = \begin{bmatrix} \mathbf{ImageMatrix1} \\ \mathbf{ImageMatrix2} \\ \dots \\ \dots \\ \mathbf{ImageMatrixN} \end{bmatrix}$$

- b. Step 2.2: Take the transpose of **Train1** and assign it to another matrix, **Train2**.

$$\mathbf{Train2} = \mathit{Transpose}(\mathbf{Train1})$$

- Step 3: For the new image to be classified,
 - a. Step 3.1: Convert the image to a matrix of length $R \times C$ and put it N times together in another big image-matrix, **Test1** like this:

$$\mathbf{Test1} = \begin{bmatrix} \mathbf{NewImageMatrix} \\ \mathbf{NewImageMatrix} \\ \dots \\ \dots \\ \mathbf{NewImageMatrix} \end{bmatrix}$$

- b. Step 3.2: Take the transpose of **Test1** and assign it to another matrix, **Test2**.

$$\mathbf{Test2} = \mathit{Transpose}(\mathbf{Test1})$$

[label=.]

- Step 4: For both big image matrices,
 - a. Step 4.1: Apply PCA
 - b. Step 4.2: Select k eigenvectors with the highest eigenvalues
- Step 5: Determine the similarity of the new image with the existing images using the k extracted features i.e., determine the similarity of **Test1** with **Train1** and **Test2** with **Train2** using the k extracted features.
- Step 6: Classify the new input image either to “face” if the similarity is highest, or to “non-face”, otherwise.

4.2. Problem statement-2: Recognition of individual face

Given a new image, classify it to most similar *image(s)* from a set of N original face images for each of the m peoples, each image is R pixels high by C pixels wide i.e., the size is $R \times C$. To solve this, we merge N training image matrices for each of the m people into a separate single big matrix by placing one after another. Then, we also place the input image matrix N times one after another to form another big matrix. After that, we take the transpose of all big matrices. Subsequently, we apply PCA on all big matrices and select k eigenvectors for each. We then determine the similarity of the normal input big matrix with all normal training big matrices, and transposed input big matrix with all transposed training big matrices using selected k features. Finally, the decision is taken based the similarity result. The solution is illustrated with the following steps:

- Step 1: Input the N original images of size $R \times C$ for each of the m peoples.
- Step 2: For each of the N images of each of the m peoples, convert it into a matrix of length $R \times C$
 - Step 2.1: Put the matrices together in a separate big image-matrix, **Train3** like this:

$$\mathbf{Train3} = \begin{bmatrix} \mathbf{ImageMatrix1} \\ \mathbf{ImageMatrix2} \\ \dots \\ \dots \\ \mathbf{ImageMatrixN} \end{bmatrix}$$

- Step 2.2: Take the transpose of **Train3** and assign it to another matrix, **Train4**.

$$\mathbf{Train4} = \mathit{Transpose}(\mathbf{Train3})$$

- Step 3: For the new image to be classified,
 - Step 3.1: Convert the image to a matrix of length $R \times C$ and put it N times together in another big image-matrix, **Test3** like this:

$$\mathbf{Test3} = \begin{bmatrix} \mathbf{NewImageMatrix} \\ \mathbf{NewImageMatrix} \\ \dots \\ \dots \\ \mathbf{NewImageMatrix} \end{bmatrix}$$

- Step 3.2: Take the transpose of **Test3** and assign it to another matrix, **Test4**.

$$\mathbf{Test4} = \mathit{Transpose}(\mathbf{Test3})$$

- Step 4: For both big image matrices,
 - Step 4.1: Apply PCA
 - Step 4.2: Select k eigenvectors with the highest eigenvalues
- Step 5: Determine the similarity of the new image with all the existing images of m peoples using the k extracted features i.e., determine the similarity of **Test3** with **Train3** and **Test4** with **Train4** using the k extracted features.

- Step 6: Classify the new input image to the most probable *image(s)* with the highest similarity.

To determine the similarity for both problem statements, first, each eigenvector in a training set is subtracted with its corresponding eigenvector in the testing set. Then the result of each eigenvector is averaged. Now, the new instance is classified as “yes”, if the average values are near to a threshold value, say α , that would be ideally around zero (0).

5. RESULTS

The proposed method for face recognition based on principal components has been implemented in MATLAB simulation platform. The implemented code has been tested for some common face images captured manually. In addition, it has been tested for the two popular face image databases: ORL and Yale. In ORL database, there are 10 different grayscale images of each of 40 distinct subjects. For some of the subjects, the images were taken at different times, and with the variation of lighting and facial expressions. All images were captured against a dark homogeneous background with the subjects in an upright, frontal position. In Yale database, there are 11 different grayscale images of each of 15 distinct subjects/individuals, one per different facial expression or configuration. Yale has extensions called Extended Yale Face Database A and B. Extended Yale Face Database B has 38 subjects/individuals and around 64 near frontal images under different illuminations per subject. For both databases, there are two types of pixel resolution for the images available: 32×32 and 64×64 . Some images from ORL and Yale and extended Yale face database B are shown in Figures 1 (a-c) respectively [25] while Table 1 shows the results on different data distributions.

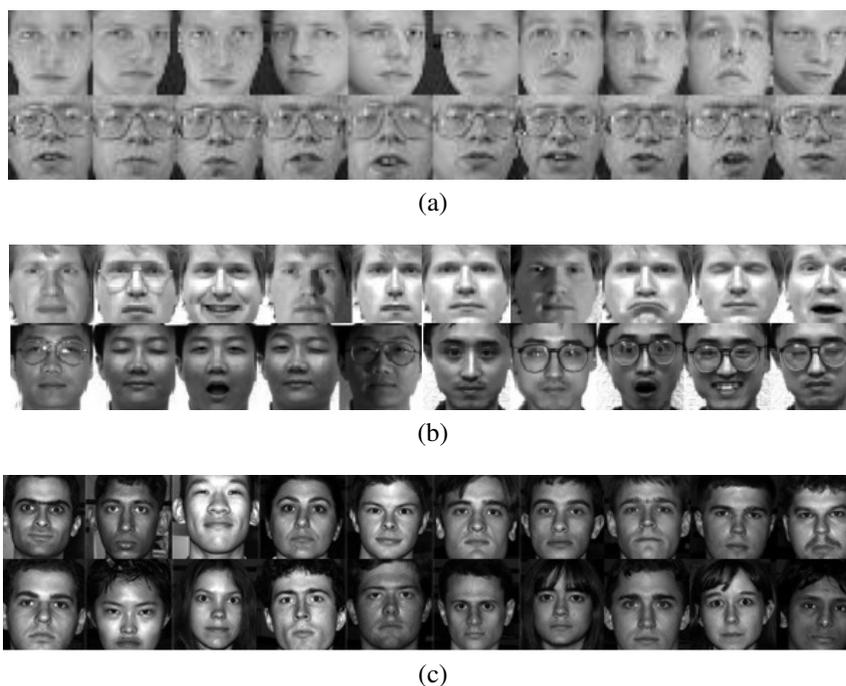


Figure 1. Face databases: (a) sample images from the ORL database, (b) sample images from the Yale database, and (c) sample images from the extended Yale face database B

For the database, the training and testing sets are created in the same manner mentioned above. For the first problem statement, a random subset of images from every subject was taken to form the training set, **Train1** and thus **Train2**. The other images were considered to be the testing set, **Test1** and thus **Test2**. For the second problem statement, a random subset of images per every subject was taken to form the training set, **Train3** and thus **Train4**. Any of the rest image(s) of the respective subject, upon which the training sets are formed, was considered to be the testing set, **Test3** and thus **Test4**. The recognition result of the

Table 1. Databases and results

Task	Database	Total number of samples	Samples of individual subject	α
Recognition of a typical face	ORL	400	40	Theoretically: 0; Practically: around 0
	Yale	2432	38	Theoretically: 0; Practically: around 0
Recognition of Individual Face	ORL	400	40	Theoretically: 0; Practically: around 0
	Yale	2432	38	Theoretically: 0; Practically: around 0

proposed method was quite acceptable because of, especially, the training sets, **Train2** and **Train4**, which are the transpose of the original training sets, **Train1** and **Train3** respectively. The recognition accuracy can significantly be decreased with the inconsistent images in the training sets.

6. CONCLUSION AND FUTURE WORK

The discussed comprehensive overview of DR techniques and the working principle of PCA can be the ingredients for developing a typical image-data mining application. The proposed method for face recognition based on principal components can, mostly, be used in those applications where a few images are enough to train. The proposed approach can be used for not only face recognition but also for other kind of objects recognition in the same manner. In future, the proposed technique will be applied on ORL and Yale databases completely along with other face databases and its performance will be compared with the existing classifiers based on either machine learning algorithms or other statistical approaches. In addition, an adaptive range of the threshold, α to recognize an instance will be determined.

REFERENCES

- [1] J. Han, J. Pei, and M. Kamber, "Data mining: concepts and techniques," *Elsevier*, 2011.
- [2] I. H. Witten and E. Frank, "Data mining: practical machine learning tools and techniques with Java implementations," *ACM SIGMOD Record*, vol. 31, no. 1, pp. 76-77, 2002, doi: 10.1145/507338.507355.
- [3] M. F. Rabbi *et al.*, "Performance Evaluation of Data Mining Classification Techniques for Heart Disease Prediction," *American Journal of Engineering Research (AJER)*, vol. 7, no. 2, pp. 278-283, 2002.
- [4] S. M. M. Hasan, M. A. Mamun, M. P. Uddin and M. A. Hossain, "Comparative Analysis of Classification Approaches for Heart Disease Prediction," *2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2)*, 2018, pp. 1-4, doi: 10.1109/IC4ME2.2018.8465594.
- [5] L. Maaten, E. Postma, and J. Heri, "Dimensionality Reduction: A Comparative Review," *Journal of Machine Learning Research*, vol. 10, no. 1, 2009.
- [6] A. W. Altaher and S. K. Abbas, "Image processing analysis of sigmoidal Hadamard wavelet with PCA to detect hidden object," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 18, no. 3, pp. 12161223, Jun. 2020, doi: 10.12928/telkomnika.v18i3.13541.
- [7] S. A. Baker, H. H. Mohammed, and H. A. Aldabagh, "Improving face recognition by artificial neural network using principal component analysis," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 18, no. 6, pp. 3357-3364, doi: 10.12928/telkomnika.v18i6.16335.
- [8] R. Kavitha and E. Kannan, "An efficient framework for heart disease classification using feature extraction and feature selection technique in data mining," *2016 International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS)*, 2016, pp. 1-5, doi: 10.1109/ICETETS.2016.7603000.
- [9] R. N. Rohmah, B. Handaga, N. Nurokhim, and I. Soesanti, "A statistical approach on pulmonary tuberculosis detection system based on X-ray image," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 17, no. 9, pp. 1474-1482, Jun. 2019, doi: 10.12928/telkomnika.v17i9.10546.
- [10] O. A. Adegbola, I. A. Adeyemo, F. A. Semire, S. I. Popoola, and A. A. Atayero, "A principal component analysis-based feature dimensionality reduction scheme for content-based image retrieval system," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 18, no. 4, pp. 1892-1896, Aug. 2020, doi: 10.12928/telkomnika.v18i4.11176.
- [11] S. Raschka, "Principal component analysis in 3 simple steps," 2015. [Online]. Available: <http://sebastianraschka.com/Articles/2015pca3steps.html>
- [12] M. Dash, H. Liu and J. Yao, "Dimensionality reduction of unsupervised data," *Proceedings Ninth IEEE International Conference on Tools with Artificial Intelligence*, 1997, pp. 532-539, doi: 10.1109/TAI.1997.632300.
- [13] A. Ghodsi, "Dimensionality reduction a short tutorial," Department of Statistics and Actuarial Science, University of Waterloo Waterloo, Ontario, Canada, pp. 1-25, 2006.

- [14] C. O. S. Sorzano, J. Vargas, and A. Pascual, "A survey of dimensionality reduction techniques," *Machine Learning*, pp. 1–35, 2014.
- [15] M. Loog, B. van Ginneken, and R. P. W. Duin, "Dimensionality reduction of image features using the canonical contextual correlation projection," *Pattern Recognition*, vol. 38, no. 12, pp. 2409–2418, 2005, doi:10.1016/j.patcog.2005.04.011.
- [16] M. G. B. Blum, M. A. Nunes, D. Prangle, and S. A. Sisson, "A Comparative Review of Dimension Reduction Methods in Approximate Bayesian Computation," *Statistical Science*, vol. 28, no. 2, pp. 189–208, May 2013, doi: 10.1214/12-STS406.
- [17] R. Silipo, I. Aadae, and A. H. M. Berthold, "Seven techniques for dimensionality reduction," Open for Innovation KNIME, 2014. [Online]. Available: https://www.knime.com/sites/default/files/inline-images/knime_seventechniquesdatadimreduction.pdf
- [18] S. K. Joshi and S. Machchhar, "An evolution and evaluation of dimensionality reduction techniques — A comparative study," *2014 IEEE International Conference on Computational Intelligence and Computing Research*, 2014, pp. 1–5, doi: 10.1109/ICCIC.2014.7238538.
- [19] W. Nick, J. Shelton, G. Bullock, A. Esterline and K. Asamene, "Comparing dimensionality reduction techniques," *SoutheastCon 2015*, 2015, pp. 1–2, doi: 10.1109/SECON.2015.7132997.
- [20] W. Huang and H. Yin, "Linear and nonlinear dimensionality reduction for face recognition," *2009 16th IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 3337–3340, doi: 10.1109/ICIP.2009.5413898.
- [21] A. Alkandari and S. J. Aljaber, "Principle Component Analysis algorithm (PCA) for image recognition," *2015 Second International Conference on Computing Technology and Information Management (ICCTIM)*, 2015, pp. 76–80, doi: 10.1109/ICCTIM.2015.7224596.
- [22] S. K. Dandpat and S. Meher, "Performance improvement for face recognition using PCA and two-dimensional PCA," *2013 International Conference on Computer Communication and Informatics*, 2013, pp. 1–5, doi: 10.1109/ICCCI.2013.6466291.
- [23] H. Su and X. Wang, "Principal Component Analysis in Linear Discriminant Analysis Space for Face Recognition," *2014 5th International Conference on Digital Home*, 2014, pp. 30–34, doi: 10.1109/ICDH.2014.13.
- [24] D. Kumar, C. S. Rai and S. Kumar, "Face Recognition using Self-Organizing Map and Principal Component Analysis," *2005 International Conference on Neural Networks and Brain*, 2005, pp. 1469–1473, doi: 10.1109/ICNNB.2005.1614908.
- [25] D. Cai, "Four face databases in matlab format." [Online]. Available: <http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html>