Facial recognition using two-dimensional principal component analysis and k-nearest neighbor: a case analysis of facial images

E Sugiharti*, A T Putra, and Subhan
Lecturers of Computer Science of UNNES

*Corresponding Author: endangsugiharti@mail.unnes.ac.id

Abstract. Science and Technology Innovation in Computer Science, especially in the facial image is increasingly needed in the Industrial Era of 4.0. The problem: How to use the working pattern of the Two-Dimensional Principal Component Analysis (2DPCA) method that integrated with K-Nearest Neighbors (KNN) in the Facial Image Recognition for various purposes? The purpose of this study, to analyze the Facial Image Recognition based on the method of Two-Dimensional Principal Component Analysis (2DPCA) which is integrated with KNN. This research uses the method of Two-Dimensional Principal Component Analysis (2DPCA) for the feature extraction process and the KNN classification method is applied to perform the data classification process so that the desired accuracy value is obtained. Research subjects use the image database from UCI repository, consists of 190 black and white face images of people taken with varying pose (straight, left, right, up), expression (neutral, happy, sad, angry), and size. The result of this study is the performance analysis of Facial Image Recognition based on the method of Two-Dimensional Principal Component Analysis (2DPCA) that is integrated with k-Nearest Neighbors (KNN).

1. Introduction
Research that has fundamental idea planned by the government through Kemenristekdikti need to be supported by Universitas Negeri Semarang (UNNES). On the other hand, science and technology innovation in the field of computers, especially Face Image Recognition is increasingly needed in the industrial era 4.0. Thus, the analysis of Facial Image Recognition based on the latest findings needs to be developed. In the current industrial era 4.0, the development of digital image processing is increasingly extensive, one of which is in the field of pattern recognition in digital images. Detection and recognition of patterns in the image are very broad and many developed with a number of approaches.

The face image recognition system is one of the biometric technology identification techniques using the individual's face as the main parameter. Pattern recognition is a method that works to find patterns in data that indicate a certain information. Pattern Recognition is a discipline that aims to classify objects into categories or classes. Patterns are defined entities and can be identified through their features.

According to Das [1-3], in general the process of facial image recognition consists of three main processes, namely: face detection, feature extraction, and face recognition. In general, methods in face recognition can be grouped into three approaches based on the data they need, namely: Holistic Approach, Feature-based Approach, Hybrid Approach.
Digital imagery is a two-dimensional representation of images as a limited set of digital values, known as image elements or pixels [4,5]. Image processing is the analysis and manipulation of digital images, especially to improve the quality of image processing. Image recognition techniques can be applied in a variety of different fields such as diagnostic image analysis, operations planning, object detection, traffic control systems, iris recognition, facial recognition and others [6,7].

As one of the biometric technologies, facial image recognition has actually been used for the past 50 years. However, because of its very large utilization, until now the technology is still being developed. Facial recognition systems require human intervention to find features, such as eyes, ears, nose and mouth, in the image. Facial image recognition is usually used in security systems and can be combined with other biometrics such as fingerprints or iris recognition systems [8]. Face image recognition system is a computer application that automatically identifies a person from an image or video database.

The methods used in facial image recognition can be classified into the feature-based method and the feature geometry method. One of the most popular image analysis applications is facial image processing. In this case, facial image recognition has a major role in security, access to personal information and better machine interaction with humans [9]. Facial recognition system consists of 3 steps namely face detection, feature extraction & face recognition.

Image representation and feature extraction are commonly used techniques in the face recognition process. 2DPCA was proposed by Yang et al [10]. 2DPCA has lower dimensions and is more efficient than PCA. 2DPCA directly extracts features from the matrix by projecting the image matrix along the projection axis which is an eigenvector of the scatter matrix [11].

2DPCA is based on 2D image matrices, so image matrices do not need to be converted into vectors before feature extraction. 2DPCA has two advantages. First, it's easier to evaluate the covariance matrix accurately. Second, a shorter time to determine the corresponding Eigen vector.

Classification is the process of finding a model (or function) that describes and distinguishes classes of data or the concept of purpose so that the model can be used to predict class objects whose labels are unknown or can predict the tendency of data that appears in the future. The classification method also aims to map data into pre-defined classes based on data attribute values.

One classification algorithm is K-Nearest Neighbor (KNN). The K-Nearest Neighbor (KNN) method was first introduced in the early 1950s [12]. The K-Nearest neighbor algorithm is the simplest classification algorithm of all machine learning algorithms.

KNN classification is done by comparing the distance between training data with testing data. When there is inputting testing data, the KNN looks for the closest distance (Euclidean distance) of testing data to known training data. Euclidean distance matrix is used to determine the proximity of data points/distance between data in K-Nearest Neighbor.

In the training phase, this algorithm only stores feature vectors and classifies training sample data. In the classification phase, the same features are calculated for testing data (whose classification is unknown). The distance of this new vector to the entire training sample vector is calculated and the closest number of k is taken. The new points are predicted to be included in the most classifications of these points.

Sugiharti [13, 14] in their research reported that technological advances in technology and information could not be dammed. Especially when it comes to programming such as face detection. Every human being is created with a different physical form so that these humans can be distinguished from one another. In information technology, the uniqueness of human physical form can be used as input to a face identification system design, authentication, or security system.

From the background description above, the problems that will be resolved through the study in this article are as follows. How to use the working pattern of the Two-Dimensional Principal Component Analysis (2DPCA) method with K-Nearest Neighbor (KNN) in facial image recognition for digital image processing?

The aim is to analyze the work patterns of facial image recognition based on the Two-Dimensional Principal Component Analysis (2DPCA) method with K-Nearest Neighbor (KNN), so that in the future
Face Image Recognition can have alternative methods and programs that make Face Image Recognition expected to be sharper, accurate, and measurable.

2. Method
Research subjects in this research use the image database from UCI Machine Learning repository, consists of 23 face images of people taken with varying pose (straight, left, right, up), expression (neutral, happy, sad, angry), eyes (wearing sunglasses or not), and size. The dataset on each face consists of 10 images, of which images 1st to 8th are the training dataset, while the 9th and 10th images are the test dataset for testing. So, the total images used are 230.

This study uses the method of Two-Dimensional Principal Component Analysis (2DPCA) for the feature extraction process and the K-Nearest Neighbor (KNN) classification method is applied to perform the data classification process so that the desired accuracy value is obtained.

The method used for facial image recognition is a combination of the Two-Dimensional Principal Component Analysis (2D-PCA) and K-Nearest Neighbor (KNN) methods. Stages of facial image processing:
1. First, the face image is extracted into $A_{mxn}$ vector matrix and then converted to the matrix $Y = AX$. The data is divided into training data and test data.
2. The next step is to calculate the average of the total training set matrices.
   $\bar{A} = \frac{1}{M} \sum_k A_k$
3. Calculates the difference matrix of each $A_k$ image with an average of the total matrix of the training set $\bar{A}$.
4. Furthermore, the covariance matrix can be calculated from the set of training images, namely by the equation $G = \frac{1}{M-1} \sum_{k=1}^{M} (A_k - \bar{A})^T (A_k - \bar{A})$. The G matrix is called the face image covariance matrix and the G matrix is evaluated with the face image from the entire training dataset.
5. Determine the eigenvalue and eigenvector of the resulting covariance matrix.
6. Training data and testing data that have been made into one matrix will be compared using the k-Nearest Neighbor method with Euclidean distance search with the equation $d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$.

3. Results and Discussion

3.1. Results
In this research, there are some steps in the analysis of image data for feature extraction and classification process.

3.1.1. Designing of system. The making of system design is used to solve the face recognition problem which the feature extraction using 2DPCA method and continue with the use of KNN for the classification. This face recognition system was designed using Python language and Django as the framework.

3.1.2. Splitting data as training data and testing data. The dataset is divided into two parts, namely training data and testing data. The dataset on each face consists of 10 images, of which figures 1 to 8 are used as a training dataset, while figures 9 and 10 are dataset datasets.

3.1.3. Features extraction. The feature extraction step in this system is carried out as follows: (a) change the face image into a matrix, (b) calculate the mean vector of a face image, (c) calculate deviation from mean, (d) calculate the covariance matrix, (e) calculate eigenvalues and eigenvectors, (f) calculate eigenfaces.
3.1.4. Classification. The classification stage is the process of matching the testing image class with the training image. The method used in this study is K-Nearest Neighbor (KNN), the first step in this classification is to find the eigenface value of the training image. The second step, calculate the euclidean distance between the training image and the testing image. The class that has the smallest euclidean distance is considered a class that has many similarities with the testing image. Furthermore, the accuracy calculation is used to find out the level of accuracy of the system in recognizing face image classes. Confusion matrix provides decisions obtained in training and testing, confusion matrix provides an assessment of the performance of classifications based on true or false objects.

The results of the calculations produce the accuracy of facial recognition, the level of accuracy, and the time required for the face recognition process, as shown in Figures 1 and Figure 2.

![Figure 1. The result of face recognition](image1)

![Figure 2. Conclusion of face recognition](image2)

**Table 1.** Test of face recognition based on eigenvalue contribution.

| No | Eigen Contribution (%) | k | Recognition Error | Accuracy (%) |
|----|-------------------------|---|-------------------|--------------|
| 1. | 5                       | 5 | 2                 | 94.74        |
| 2. | 10                      | 5 | 2                 | 94.74        |
| 3. | 30                      | 5 | 2                 | 94.74        |
| 4. | 35                      | 5 | 7                 | 81.58        |
| 5. | 50                      | 5 | 7                 | 81.58        |
| 6. | 65                      | 5 | 9                 | 76.32        |
| 7. | 70                      | 5 | 10                | 73.68        |
| 8. | 75                      | 5 | 8                 | 78.95        |
| 9. | 95                      | 5 | 2                 | 94.74        |
| 10.| 100                     | 5 | 2                 | 94.74        |

**Table 2.** Test of face recognition based on eigenvalue contribution of 30%.

| No | K | Recognition Error | Accuracy (%) |
|----|---|-------------------|--------------|
| 1. | 1 | 1                 | 97.37        |
| 2. | 2 | 1                 | 97.37        |
| 3. | 3 | 1                 | 97.37        |
| 4. | 4 | 2                 | 94.74        |
| 5. | 5 | 2                 | 94.74        |
| 6. | 6 | 2                 | 94.74        |
| 7. | 7 | 4                 | 89.47        |
| 8. | 8 | 4                 | 89.47        |
| 9. | 9 | 5                 | 86.84        |
| 10.| 10| 6                 | 84.21        |
A face recognition test is performed on the face image taken from the UCI machine learning repository dataset based on the contributions of eigenvalues (in %), the results of which can be seen in Table 1. After that, the testing also done by taking a certain percentage of eigenvalue with different values of $k$ as shown in Table 2 and Table 3.

**Tabel 3.** Test of face recognition based on eigenvalue contribution of 95%.

| No | $K$ | Recognition Error | Accuracy (%) |
|----|-----|-------------------|--------------|
| 1  | 1   | 1                 | 97.37        |
| 2  | 2   | 1                 | 97.37        |
| 3  | 3   | 3                 | 92.11        |
| 4  | 4   | 2                 | 94.74        |
| 5  | 5   | 2                 | 94.74        |
| 6  | 6   | 2                 | 94.74        |
| 7  | 7   | 3                 | 92.11        |
| 8  | 8   | 4                 | 89.47        |
| 9  | 9   | 4                 | 89.47        |
| 10 | 10  | 4                 | 89.47        |

3.2. Discussion

This study uses face image data from the UCI Machine Learning Repository which consists of 190 face images. UCI Machine Learning repository is one of the benchmark datasets [15]. The dataset is divided into training data and testing data in accordance with predetermined compositions used in data processing. The system design is used to assist in solving the face recognition problems using the method of Two-Dimensional Principal Component Analysis (2DPCA) for feature extraction and K-Nearest Neighbor (KNN) for classification.

This face recognition application using 2DPCA with KNN was built using Python Language by utilizing the Django framework. The initial part of the application starts with the appearance of the login page for the user. To input data, click the KNN menu and then click the Data sub menu. Then it will be directed to the application dataset display. After determining which dataset to use, the dataset will be read by the application marked by counting the amount of data. After the data has been uploaded successfully, then click the Results sub-menu to arrange training for KNN. Enter the eigenvalue parameters, $k$, and the number of images that used as the training images in each data folder.

The results of calculations in the form of the accuracy of facial recognition, the level of accuracy, and the time required to do the facial recognition process. The face recognition test on the UCI Machine Learning Repository face image based on the contribution of eigenvalues (in %) gives the result that the best recognition is in the contribution of eigenvalues of 5%, 10%, 30%, 95%, and 100% which in that values the error of face recognition is the lowest one, and also marked by the highest recognition accuracy is 94.74%.

Furthermore, the influence of $k$ value to the level of facial recognition will be analyzed by taking an eigenvalue of 30% and 95%. The face recognition test results on the face image of UCI Machine Learning Repository with a contribution of 30% of eigenvalue obtained the best results at $k = 1$, $k = 2$, and $k = 3$ with the smallest recognition error compared to other $k$ values. This is also indicated by the highest level of accuracy which is 97.37%. While the test with the contribution of 95% of eigenvalue obtained the best results at values $k = 1$ and $k = 2$, with the smallest recognition error compared to other $k$ values. This is also indicated by the highest level of accuracy which is 97.37%.

From the various tests above, it can be seen that face recognition using 2DPCA and KNN methods is affected by the contribution of the eigenvalue used and also supported by the $k$ value. The right combination of these two parameters will provide the best face recognition image. In addition, image size and resolution also affect the level of face image recognition, where facial image recognition errors can occur if the test image size and training image are different.
4. Conclusion

The conclusion is as follows. Facial recognition using 2DPCA and KNN methods is influenced by the contribution of the eigenvalues used and also influenced by the $k$ value. The right combination of the two parameters will provide the best level of face image recognition. Besides, the consistency of image size and resolution also affects the level of facial image recognition.

References

[1] Das DK 2012 Int. J. Emerg. Technol. Adv. Eng. 2 330
[2] Erwin et al 2019 J. Phys.: Conf. Ser. 1196 012009
[3] Zeng N, Zhang H, Song B, Liu W, Li Y, and Dobaie AM 2018 Neurocomputing 273 643
[4] Li Z 2019 J. Vis. Commun. Image Represent. 65 102689
[5] Soni S & Sahu RS 2013 Int. J. Adv. Res. Comput. Commun. Eng. 2 3899
[6] Mainzer K, Killinger S, McKenna R, and Fichtner W 2017 Sol. Energy 155 561
[7] Du H, Hu Q, Jiang M, & Zhang F 2015 J. Vis. Commun. Image Represent 32 55
[8] Gosavi AP and Khot SR 2013 Int. J. Soft Comput. Eng 3 258
[9] Hidayah MM, Akhlis I, & Sugiharti E 2017 Sci. J. Inform. 4 66
[10] Yang J, Zhang D, Frangi AF, & Yang JY 2004 IEEE Trans. Pattern Anal. Mach. Intell. 25 131
[11] Nikdel H, Forghani Y, and HMS Mohammad 2017 Expert Syst. 35 1
[12] Patil SA & Deore PJ 2013 Inform. Eng. Int. J. 1 31
[13] Sugiharti E, Firmansyah S, dan Devi FR 2017 J. Theor. Appl. Inf. Technol. 95 902
[14] Sugiharti E, Arifudin R, dan Putra AT 2018 J. Theor. Appl. Inf. Technol. 96 7176
[15] Misir R, Samanta RK 2007 Int. J. Comput. Sci. Eng. 5 40