Improvement of One-Dimensional Fisherface Algorithm to extract the Features (Case study: Face Recognition)

A Muntasa¹, M Yusuf¹, M Syarief¹, Yulmaini², P P Sari³
¹Dept. of Informatics Engineering, University of Trunojoyo Madura, Bangkalan, Indonesia
²Darmajaya Informatics and Business Institute (IBI Darmajaya) Lampung, Indonesia
³BPSDMP Ministry of Comunication and Informatics Surabaya, Surabaya, Indonesia
arifmuntasa@trunojoyo.ac.id

Abstract. Recently, computer vision research results have supported many sectors to assist and solve problems. One of branch of the computer vision fields is biometric system. Many modalities have been implemented to depict the human characteristics. Face is one of the modalities that has been employed to recognize the human. A crucial problem of the face recognition is high dimensionality. The problem would impact on the computational performance, and even it could cause the process failure. Feature extraction is the solution to reduce the dimensionality. However, many cases have shown that feature extraction could fail as singularity problem. In this research, we proposed the improvement of the fisherface algorithm to solve the singularity problem. We have modified the singularity covariance matrix so that the matrix can be further handled and processed. The purpose of the paper is to improve the performance of the fisherface algorithm. We have verified our proposed algorithm by using the Olivetty Research Laboratory face image. We applied 7-cross validations to evaluate our proposed algorithm, the evaluation results achieved more than 92% accuracy.

I. Introduction

In recent years, internet of thing has been developed to support the human need. Biometric is one of fields that have industrialized to support security system, i.e. personal, corporate, military, and also government. Face is one of superlative instrument that have been utilized to recognize the human identity. One of the most crucial problem of the face recognition is high dimensionality, where it will take longer time to attain the features. Many researchers have proposed algorithms to improve the performance of the feature extraction. Principal Component Analysis (PCA) is the first approach to reduce the image dimensionality. PCA is able to re-construct the sample sets so that they are similar to the original image [1]–[8]. However, PCA is not able to work perfectly when amount of the sample data is more than the amount of the image dimensionality, it is well known as small sample size problem (SSS) [9]–[13]. Sometimes, PCA also delivered a singular covariance matrix, so that it cannot be advance processed to produce the eigenvalues and eigenvectors.

To increase the recognition accuracy, Linear Discriminant Analysis (LDA) was proposed, which can be used to separate the different person and recognize the same class [6], [14]–[21]. The LDA is able to maximize and minimize the distance of the different and the same classes. The model of the LDA is able optimally work under ensuing two pre-requisite. Firstly, global and local data structures have been consistent. Secondly, sample sets used have distributed by using Gaussian. If two instruments cannot be delivered, then performance of the LDA results could not be improved, though the data has been normalized.

We proposed algorithm to optimize the covariance matrix value, i.e. the values of between class scatter matrixes are increased, while the values of within class scatter matrix is decreased. Optimization of the covariance matrix value can be completed by iterative process. We also proposed to change the image input into diagonal model. It is directed to reduce the effect of pose, illumination, and expression.
We compose the paper as follows, firstly, introduction as explained in this section. Secondly is proposed algorithm, where we define in detail the sequential processes to describe some stages. Thirdly, experimental, discussion, and analysis, where we also describe face image used to assess performance of the proposed algorithm. Summarize of the results is written on the conclusion section.

II. Proposed Algorithm

We proposed new algorithm to modify the previous algorithm, where new algorithm is the modification results of the Linear Discriminant Analysis algorithm. Our algorithm is not started by Principal Component Analysis, but we compute the average of the class and all data sets to gain the covariance matrix. However, our proposed algorithm is not able to work perfectly, when the covariant matrix results are closed to zero. Therefore, we need to add process to anticipate the singularity case. We add special value on the main diagonal, when the singularity case is appeared. Singularity case can appear when the determinant value of the matrix is closed to zero. Framework of our algorithm can be found in Figure 1.

We employed the sample set of the \( f(x, y) \) image, number of samples and poses for training are denoted by \( s \) and \( p \). If width and height of an image sample are designated by \( w \) and \( h \), then the image dimensionality is calculated by \( n = h \times w \). The image is also written by using the following matrix

\[
f(x, y) = \begin{pmatrix}
f(x_1, y_1) & f(x_1, y_2) & \cdots & f(x_1, y_w) \\
f(x_2, y_1) & f(x_2, y_2) & \cdots & f(x_2, y_w) \\
\vdots & \vdots & \ddots & \vdots \\
f(x_h, y_1) & f(x_h, y_2) & \cdots & f(x_h, y_w)
\end{pmatrix}
\]  

Before diagonal building process, an Equation (1) must be duplicated and placed below the original equation as written in Equation (2)

\[
f_D(x, y) = \begin{pmatrix}
f(x_1, y_1) & f(x_1, y_2) & \cdots & f(x_1, y_w) \\
f(x_2, y_1) & f(x_2, y_2) & \cdots & f(x_2, y_w) \\
\vdots & \vdots & \ddots & \vdots \\
f(x_h, y_1) & f(x_h, y_2) & \cdots & f(x_h, y_w)
\end{pmatrix}
\]

\[
f_D(x, y) = \begin{pmatrix}
f(x_1, y_1) & f(x_1, y_2) & \cdots & f(x_1, y_w) \\
f(x_2, y_1) & f(x_2, y_2) & \cdots & f(x_2, y_w) \\
\vdots & \vdots & \ddots & \vdots \\
f(x_{h+1}, y_1) & f(x_{h+1}, y_2) & \cdots & f(x_{h+1}, y_w)
\end{pmatrix}
\]

The first column matrix element is taken starting from the first until \( h \) row, the second column matrix element starts from the second until \((h+1)^{th}\) row. Retrieving matrix elements will be completed until the last column, where the last column starts from \( w^{th} \) until \((h+w-1)^{th}\) row. The diagonal matrix result is written as follows

\[
f_L(x, y) = \begin{pmatrix}
f(x_1, y_1) & f(x_2, y_2) & \cdots & f(x_w, y_w) \\
f(x_2, y_1) & f(x_2, y_2) & \cdots & f(x_{w+1}, y_w) \\
\vdots & \vdots & \ddots & \vdots \\
f(x_h, y_1) & f(x_1, y_2) & \cdots & f(x_h, y_{w-1})
\end{pmatrix}
\]
Equation (4) represents diagonal matrix of the image as written in Equation (1). Furthermore, Equation (4) is transformed into one dimensional matrix and arranged horizontally. It shows that an image has row matrix with \( n \) column, where \( n \) represents image dimension, i.e. \( n = h \times w \). As we mentioned above that the samples and poses are represented by \( s \) and \( p \). If the training sets are composed based on the row, then the training matrix has \( s \times p \) (i.e. \( m \)) as row and \( h \times w \) (i.e. \( n \)) as columns. We proposed the Linear Discriminant Analysis algorithm without Principal Component Analysis process. We have modified the process to produce non-singularity covariance matrix. We have processed iteratively the covariance matrix. The iterative process is used to keep distance between classes and close the distance between samples in one class.

Figure 1. Proposed Algorithm to Obtain the Weight and Projection Matrixes
The average for the training sets \((f_c)\) is shown in Equation (6), where the symbols of \(j\) states the image dimensionality index, i.e. \(j \in 1 \ldots n\). Unlike eigenface, fisherface needs the average of class, where it is computed based on the member of the class as shown in Equation (7). In this equation, \(c_i\) denotes \(i^{th}\) class, \(nc_i\) states number of memberships on the \(i^{th}\) class

\[
\frac{\sum_{k=1}^{s} f_k(x_{ijy})}{s}
\]

Moreover, between and within class scatter \((S_B\) and \(S_W\)) is obtained by using the simplest operation as follows

\[
S_B = \left( \left( \bar{f}_j \right)_{c_i} - \bar{f}_j \right)^T \left( \left( \bar{f}_j \right)_{c_i} - \bar{f}_j \right)
\]

\[
S_W = \left( f_c(x,y) - \left( \bar{f}_j \right)_{c_i} \right)^T \left( f_c(x,y) - \left( \bar{f}_j \right)_{c_i} \right)
\]

The results show that, sometimes determinant of \(S_B\) or \(S_W\) produces a value close to zero, it is call as singularity problem. To anticipate this case, we proposed to improve it, where the main diagonal elements are added the \(\partial\) value, i.e. \(0 < \partial < 1\) as shown in Equation (10) and (11)

\[
S_B = \left\{ S_B(k,k) + \partial, \ \det(S_B) = 0 \right\}
\]

\[
S_W = \left\{ S_W(k,k) + \partial, \ \det(S_W) = 0 \right\}
\]

The eigenvalue and eigenvector can be easily found in Equation (13)

\[
\lambda = S_B \times (S_W)^{-1}
\]

The results of the Equation (13) are eigenvalues(\(\lambda\)) and eigenvectors(\(A\)). However, the eigenvalues are necessary to be decreasingly ordered and followed by matching columns of the eigenvector values [1], [17], [18], [22]. The larger of the eigenvalue showed the more significant of the characteristic. Furthermore, the projection (\(\varphi\)) and weight (\(W\)) matrices can be obtained by using the sample sets and the eigenvector values

\[
\varphi = \left( \left( f_c(x,y) \right)^T \times \Lambda \right)^T
\]

\[
W = \left( f_c(x,y) \right) \times \varphi^T
\]

Superscript of \(T\) states transpose matrix, while subscript -1 denotes invers matrix. The weight of the testing sample is generated by training sets and transpose of the projection matrix as follows

\[
Z = F(x,y) \times \varphi^T
\]

Performance of the proposed algorithm can be measured by using Euclidian Distance and Manhattan algorithms as follows

\[
D_E = \| W - Z \|
\]
\[ D_H = |W - Z| \]  \hspace{1cm} (18)

If the testing image samples evaluated is symbolized by using \( R \), and number of true classifications is represented by using \( \mathbb{I} \), and \( \mathbb{I} \leq R \), then the accuracy of the classification could be expressed easily by using the equation below

\[ A = \frac{\mathbb{I}}{R} \times 100\% \] \hspace{1cm} (19)

### III. Experimental, Discussion, and Analysis

We have carried out many experiments to verify our proposed algorithm. We have employed the Olivetty Research Laboratory (ORL) to measure our proposed algorithm. The ORL has been apprehended 40 people. The people have been captured with different poses, where a person has 10 different poses. The original image has 92 x 112 image pixels [23]. We have organized the image samples into two parts, i.e. training sets and testing sets. Figure 2 shows the ORL sample. We have selected two people as image samples, where they consist of 5 poses as shown in Figure 2.

![Figure 2](image.png)

**Figure 2. The ORL Image Samples**  
(2 People, for each person taken 5 poses)

In this paper, the proposed algorithm has scrambled the training sets for four times. We employed three experimental models, i.e. two, three, and four images as training sets while the rest of images as the testing sets (eight, seven, and six images) as shown in Table 1, 2, and 3.

In Table 1, we applied two images as the training, it means the rest of eight images will be utilized as the testing. If the ORL has 40 classes/people, then our proposed algorithm has evaluated 240 images. If an image is also tested by using the different features (16 until 24 different features), then experiment 1 is evaluated 5,760 times (240 images, an image is evaluated by using 20 different scenarios, 240x24=5,760). If Table 1 has 7 experiments with different index poses (in this case, the system has scrambled two of ten images as the training samples. It is applied four times), then number of image evaluations is 23,040 times (5,760x4=23,040).

| No  | Four-Cross validations | Features Selection |
|-----|------------------------|--------------------|
|     |                        | 1st Scenario       | 24th Scenario      |
| 1   | Cross Validation 1     | 1-16                | …                  |
| 2   | Cross Validation 2     | 1-16                | …                  |
| 3   | Cross Validation 3     | 1-16                | …                  |
| 4   | Cross Validation 4     | 1-16                | …                  |

In Table 2, where in this table we employed three images as the training and the rest of the image as the testing. We have decreased number of the training samples, but the number of the features used is decreased. In this case, we employed 40 different images, 7 poses, and 20 scenarios. It shows that we have evaluated 6,720 for each experiment. If four experiments have been applied, then the evaluation has been conducted 26,880 times.

Similar model could be shown in Table 2, where in this table we employed three images as the training and the rest of the image as the testing. We have decreased number of the training samples, but the number of the features used is decreased. In this case, we employed 40 different images, 7 poses, and 20 scenarios. It shows that we have evaluated 6,720 for each experiment. If four experiments have been applied, then the evaluation has been conducted 26,880 times.
Table 2. Second Experiment Model
(Three and Seven Images as the Training and Testing Samples)

| No | Four-Cross validations | Features Selection | 1<sup>st</sup> Scenario | 20<sup>th</sup> Scenario |
|----|-------------------------|--------------------|--------------------------|--------------------------|
| 1  | Cross Validation 1      | 1-11               | ...                      | 1-30                     |
| 2  | Cross Validation 2      | 1-11               | ...                      | 1-30                     |
| 3  | Cross Validation 3      | 1-11               | ...                      | 1-30                     |
| 4  | Cross Validation 4      | 1-11               | ...                      | 1-30                     |

We also decrease the number of features involved on the classification, but the number of training sets involved is increased, i.e. four images. The remaining of the image for the testing is six of ten images. If the people involved is forty, number of testing is six, number of scenarios is fifteen, number of experiments is four, then number of evaluations is 14,400 times (40x6x15x4=14,400).

Table 3. Third Experiment Model
(Four and Six Images as the Training and Testing Samples)

| No | Four-Cross validations | Features Selection | 1<sup>st</sup> Scenario | 15<sup>th</sup> Scenario |
|----|-------------------------|--------------------|--------------------------|--------------------------|
| 1  | Cross Validation 1      | 1-6                | ...                      | 1-25                     |
| 2  | Cross Validation 2      | 1-6                | ...                      | 1-25                     |
| 3  | Cross Validation 3      | 1-6                | ...                      | 1-25                     |
| 4  | Cross Validation 4      | 1-6                | ...                      | 1-25                     |

All of the results have been described in Figure 3, 4, and 5. The word of CV-1 states the first cross validation, similarly CV-2, CV-3, and CV-4 represents the second, the third, and the last cross validation respectively. The first experiment model has delivered the results, the accuracy is computed based on the last equation, which is Equation (19). The accuracy results have a tendency to increase the accuracy, when the features involved are also increased as shown in Figure 3. The performance of the proposed algorithm has obtained 85.94% as the highest accuracy, it is depicted on the first cross validation when it uses 1 until 28 features and 1 until 39 features. We have investigated the difference results for each cross-validation, where the difference is represented by using standard deviation, i.e. 1.47%, 1.50%, 1.63%, and 1.63% for the first, second, third, and the last cross validation respectively. It indicated that the difference of the accuracy results is small. Another indication could be described by using the average of the accuracy for each cross validation, i.e. 83.18%, 83.28%, 82.94%, and 82.86% for the first, second, third, and the last cross validation respectively.

Figure 3. First Experiment Model Results
(Two and Eight Images as the Training and Testing Samples)
We have increased the image samples, when training process is carried out, i.e. three images as the training samples, but we reduced the features, where we started to involve eleven until thirty features as depicted in Figure 4. Overall, the average of accuracy is 90.71%, 91.43%, 91.07%, and 91.07% for the first, second, third, and the last cross validation. The second experimental model results are higher than the first experimental model results. It could be shown by the increasing of the average of the accuracy for each cross validation. The difference of the results for each cross validation could be described by the values of the standard deviation, i.e. 0.47%, 0.65%, 0.46%, and 0.27% for the first until the last cross validation. It is clear that, the second experimental has produced the better results than the first experimental. The second cross validation has delivery the highest accuracy, which is 91.43%.

The last experiments, we also increased the training samples that have been involved for the training process, contrary we decreased the features that have been applied to classify the object. In this case, we employed four images to obtain the features, but we only implemented six until twenty-five features to characterize the face image as shown in Figure 5. The average of the cross validations result delivered 92.69%, 91.92%, 93.04%, and 92.73% accuracies for the first until the last cross validation, while standard deviation obtained is 0.50%, 0.67%, 0.70%, and 1.48% for the first, second, third, and fourth cross validation respectively. Based on the accuracy results, the last experimental model has obtained the better results than the first and second experimental model, but the second experimental model has produced more stable accuracy than the first and the last experimental model as shown standard deviation. The maximum accomplishment of the last experimental model is 93.75 accuracy as shown the third cross validation in Figure 5.

Overall, we found singularity problems on the first, second, and the last experiment model, when we employed the classical fisherface algorithm. However, the problems could not give the solution of the features required. Our proposed algorithm could answer the problem of the singularity matrix, so that the problem can be solved to produce the features required.
IV. Conclusion

We have investigated some experimental processes and results. Experimental process has produced singularity matrix when the classical fisherface is implemented. We have re-solved the singularity problem by using our proposed algorithm. The experimental results also produced high accuracy and small deviation standard. Our proposed algorithm has yielded maximum accuracy for the first, second, and the last experimental models, i.e. 85.94%, 91.43%, and 93.04%. The difference of the experimental results is also small. The smallest standard deviation is 1.47%, 0.27%, and 0.50%, it proved that our proposed algorithm can efficiently work, though the training sample has been scrambled.

References

[1] A. Singh and S. Kumar, “Face Recognition Using PCA and Eigen Face Approach,” Department of Computer Science and Engineering National Institute of Technology Rourkela, May 2012.
[2] S. M. Holland, “Principal components analysis (PCA),” Dep. Geol. Univ. Georg. Athens, GA, pp. 30602–2501, 2008.
[3] U. C. Turhal and A. Duysak, “Cross grouping strategy based 2DPCA method for face recognition,” Appl. Soft Comput. J., vol. 29, pp. 270–279, 2015.
[4] Z. Zhang and S. Liu, “Coupled principal component analysis based face recognition in heterogeneous sensor networks,” Signal Processing, vol. 126, pp. 134–140, 2016.
[5] K. Huang, “Principal Component Analysis in the Eigenface Technique for Facial Recognition,” Trinity College, 2012.
[6] J. R. Price and T. F. Gee, “Face recognition using direct, weighted linear discriminant analysis and modular subspaces,” Pattern Recognit., vol. 38, no. 2, pp. 209–219, 2005.
[7] H. Konga, L. Wang, E. K. Teoh, and X. Li, “Generalized 2D principal component analysis for face image representation and recognition*,” Neural Networks, vol. 18, no. 5, pp. 585–594, 2005.
[8] G. Anbarjafari, “Face recognition using color local binary pattern from mutually independent color channels,” Eurasip J. Image Video Process., vol. 2013, pp. 1–11, 2013.
[9] J. Liu and S. Chen, “Discriminant common vectors versus neighbourhood components analysis and Laplacianfaces: A comparative study in small sample size problem,” Image Vis. Comput., vol. 24, no. 3, pp. 249–262, Mar. 2006.
[10] M. Yang, X. Wang, G. Zeng, and L. Shen, “Joint and collaborative representation with local adaptive convolution feature for face recognition with single sample per person,” Pattern Recognit., vol. 66, no. November 2016, pp. 117–128, 2017.
[11] F. Wang, J. Wang, C. Zhang, and J. Kwok, “Face recognition using spectral features,” Pattern Recognit., vol. 40, no. 10, pp. 2786–2797, 2007.
[12] Y. Zhu, J. Liu, and S. Chen, “Semi-random subspace method for face recognition,” Image Vis. Comput., vol. 27, no. 9, pp. 1358–1370, 2009.
[13] A. Sharma and K. K. Paliwal, “A two-stage linear discriminant analysis for face-recognition,” Pattern Recognit. Lett., vol. 33, no. 9, pp. 1157–1162, 2012.
[14] J. Zhao, P. L. H. Yu, L. Shi, and S. Li, “Separable linear discriminant analysis,” Comput. Stat. Data Anal., vol. 56, no. 12, pp. 4290–4300, 2012.
[15] P. Huang, T. Li, Z. Shu, G. Gao, G. Yang, and C. Qian, “Locality-regularized linear regression discriminant analysis for feature extraction,” Inf. Sci. (Ny.), vol. 429, pp. 164–176, 2018.
[16] M. S. Mahanta and K. N. Plataniotis, “2DLDA as matrix-variate formulation of a separable 1DLDA,” Pattern Recognit. Lett., vol. 68, pp. 169–175, 2015.
[17] R. Fisher, L. D. A. Multi, K. F. D. Classification, K. F. D. A. See, and R. External, “Kernel Fisher discriminant analysis Kernel trick with LDA,” pp. 1–6, 2016.

[18] J. Ye and Q. Li, “A two-stage linear discriminant analysis via QR-decomposition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 6, pp. 929–941, 2005.

[19] J. Yang, D. Zhang, X. Yong, and J. Y. Yang, “Two-dimensional discriminant transform for face recognition,” *Pattern Recognit.*, vol. 38, no. 7, pp. 1125–1129, 2005.

[20] J. G. Wang, E. Sung, and W. Y. Yau, “Incremental two-dimensional linear discriminant analysis with applications to face recognition,” *J. Netw. Comput. Appl.*, vol. 33, no. 3, pp. 314–322, 2010.

[21] C. H. Park and H. Park, “Nonlinear discriminant analysis using kernel functions and the generalized singular value decomposition,” *SIAM J. Matrix Anal. Appl.*, vol. 27, no. 1, p. 87, 2005.

[22] J. Yang, D. Zhang, A. F. Frangi, and J. Yang, “Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, 2004.

[23] “ORL, Research Center of Att, UK, Olivetti-Att-ORL, Face Image Database,” *FaceDatabase, http://www.uk.research.att.com/facedatabase.html*. 