Eigenface Algorithm-Based Facial Expression Recognition in Conversations - An Experimental Study

Zixiang Fei[1[0000-0003-3692-3467]], Erfu Yang*[1[0000-0003-1813-5950]], David Li[2[0000-0002-6401-4263]], Stephen Butler[3[0000-0002-2103-0773]], Winifred Ijomah[1[0000-0002-0652-1486]], Neil Mackin[4[0000-0002-5415-1395]]

[1] Department of Design, Manufacture and Engineering Management
University of Strathclyde, Glasgow G1 1XJ, UK
{zixiang.fei, erfu.yang, w.l.ijomah}@strath.ac.uk
[2] Strathclyde Institute of Pharmacy & Biomedical Sciences
University of Strathclyde, Glasgow G4 0RE, UK
david.li@strath.ac.uk
[3] School of Psychological Sciences and Health
University of Strathclyde, Glasgow G1 1QE, UK
stephen.butler@strath.ac.uk
[4] Capita plc, London SW1H 0XA, UK
Neil.Mackin@capita.co.uk

Abstract. Recognizing facial expressions is important in many fields such as computer-human interface. Though different approaches have been widely used in facial expression recognition systems, there are still many problems in practice to achieve the best implementation outcomes. Most systems are tested via the lab-based facial expressions, which may be unnatural. Particularly many systems have problems when they are used for recognizing the facial expressions being used during conversation. This paper mainly conducts an experimental study on Eigenface algorithm-based facial expression recognition. It primarily aims to investigate the performance of both lab-based facial expressions and facial expressions used during conversation. The experiment also aims to probe the problems arising from the recognition of facial expression in conversations. The study is carried out using both the author's facial expressions as the basis for the lab-based expressions and the facial expressions from one elderly person during conversation. The experiment showed a good result in lab-based facial expressions, but there are some issues observed when using the case of facial expressions obtained in conversation. By analyzing the experimental results, future research focus has been highlighted as the investigation of how to recognize special emotions such as a wry smile and how to deal with the interferences in the lower part of face when speaking.

Keywords: Facial expression recognition, Eigenface algorithm, Facial expressions in conversations

1 Introduction

Facial expressions play an important role in communicating with other people, expressing feelings, emotions and can be argued to be one of the most salient social signals in the visual world. Recognizing facial expressions is important in many fields such as computer-human interface.

Machine learning has many good applications in the areas such as medical and data processing. Face feature analysis is a hot topic in machine learning and image processing [1, 2]. In this paper, a large amount of images acquired from the facial expressions of the author and one elderly person are used in the training stage, which includes lab-based facial expressions and facial expressions generated during conversations. By training the given images of facial expressions, the system which is originally written by Md. Iftekhar Tanveer will be able to recognize typical emotional facial expressions such as neutral, happy, angry, sad and surprise [3]. In this system, the Eigenface algorithm is adopted since it can reduce the size of the training images and improve efficiency in the training stage.

Nowadays, there are many algorithms for facial expression recognition and researchers continue doing the research in this topic. For example, Fathallah et al used deep learning for facial expression recognition and Jun et al used 28 facial feature key-points in images detection and Gabor wavelet filter for facial expression recognition [4, 5]. However, many existing approaches have weaknesses in dealing with complex backgrounds and head motions. There are also several good survey papers which summarize the advantages and drawbacks for recognizing facial expressions [6][7][8][9][10]. For instance, Fasel et al discuss issues relating to current facial expression recognition systems [7].

Among the many approaches available, the Eigenface algorithm has proven to be the effective methods for facial expression recognition. Originally proposed by Turk and Pentland [11], this algorithm has been employed by several researchers to deal with other related problems [12][13][14][15].

Hok-chun [12] employed the Eigenface algorithm to recognize facial expression in performance animation. In this approach, the process of recognizing facial expression is achieved by matching the input images with the images for a variety of facial expression in a database. Its simplicity allows the system to work in real time.

In a different approach, Suranga et al employed the Eigenface algorithm to both recognize different facial features and assign descriptive words to each feature [13]. In this approach, after the training stage, the system is able to match the input image with a pattern in the training set which can be described as the best match. The experiment showed that the current system is able to recognize some well-defined features like eyes, but it seemed that it was unable to cover all the features in a face.

In related research, Mohamed et al used the Eigenface technique to attempt to match the input face images with face images in the database with known identity [14]. The system mainly involved three parts, i.e., Generating Eigenfaces, Face Classification and Face Identification. In addition, Chung-Hsien researched the recognition of facial expression by the Eigenface algorithm where it was focused on speaking effect removal [15].

This paper will mainly present an experimental study employing the Eigenface algorithm-based facial expression recognition approach in the Matlab environment. In addition, the Eigenface algorithm will be particularly studied using conversational facial expressions. This paper aims to probe the problems that arise when attempting to recognize facial expressions during conversations using the Eigenface algorithm.

This paper is organized as follows. Section 1 has a general overview and sets out the outline of the paper. Subsequently, fundamental knowledge on Eigenface algorithm and principal components analysis is introduced in Section 2. Section 3 outlines the main processes required for the implementation of the algorithm in a Matlab environment. The experiments and discussions are provided in Section 4. Section 5 provides a summary and conclusion.

2 Introduction to Principal Components Analysis and Eigenface Conversion

2.1 Principal Components Analysis

Recognition of facial expressions using the Eigenface algorithm requires several steps. In these steps, the principal components analysis (PCA) is considered to be fundamental for Eigenface algorithm. In Matlab, the following main steps are required:

1. Obtain one set of random data sets
2. Adjust the dataset and minus the dataset with the average value
3. Calculate the Covariance of the adjusted dataset
4. Calculate the eigenvector, and eigenvalue of the covariance
5. Obtain the feature vector by choosing some larger eigenvalue
6. Obtain the final data by multiply the adjusted data with Feature Vector

2.2 Eigenface Conversion

The classic Eigenface algorithm contains the following steps. First, the dataset for both training and testing obtained, employing various facial expressions, is used as input. Next, the adjusted dataset is obtained by the following step: take the average image for facial expressions in the dataset as (1); minus the origin dataset with the average image as (2):

$$\psi = \frac{1}{M}\sum_{i=1}^{M} T_i, \tag{1}$$

$$\phi_i = T_i - \psi. \tag{2}$$

Then, the eigenvector and eigenvalue of the covariance of the matrix of the adjusted image dataset are obtained in (3):

$$u_l = \sum_{k=1}^{M} v_{lk}\phi_k\,, l = 1 \dots\dots M. \tag{3}$$

Also, the weight of each input image is defined below in (4):

$$\omega_k = u_k^T(T_i - \psi), k = 1 \ldots \ldots M. \quad (4)$$

After that, only major weights are selected, and the others are ignored. Both training and testing datasets are projected to the face subspace. Finally, they are compared to give the result.

3 Implementation of Eigenface Conversion

The implementation of the Eigenface conversion in Matlab is shown in Fig. 1. Many functions in Matlab are used, allowing the Eigenface conversion to be simplified to some extent. The detailed steps are as follows:

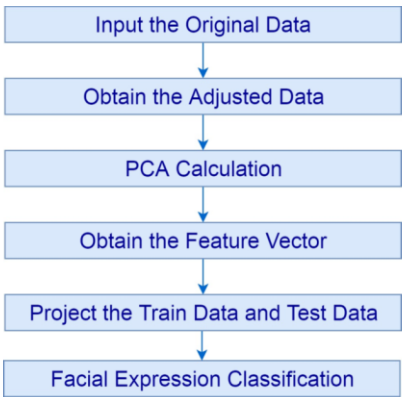

Fig. 1. Flowchart for the Eigenface Conversion

(1) Data Input
In this step, the dataset of both the training and testing phase for various facial expressions are used as input. Note that the actual facial expression for each image in the training dataset is also known.
(2) Obtain the Adjusted Dataset
Next, the adjusted dataset is obtained as follows: take the average image for facial expressions; minus the origin dataset with the average image.
(3) Obtain the Feature Vector
Then, the PCA calculation is performed to obtain the eigenvector of the covariance. The feature vector is also obtained by choosing some larger eigenvalue. There are 23, 17 and 20 images in the training dataset in the three experiments respectively. 8 largest eigenvalues are chosen in the both three experiments.
(4) Project the Training and Testing Data
In this step, the training image dataset and the testing image dataset are projected to the Facespace. They are obtained by multiplying the adjusted dataset to the feature vector.

(5) Compare and Classify the Facial Expression
In addition, the test images of facial expression are compared with the ones in the training dataset by calculating the Euclidean distance. It is assumed that the testing image has the same facial expression as the training image that has the shortest Euclidean distance. As a result, the facial expression is recognized.
(6) Generate Results
Finally, the results of recognizing facial expression are generated in a file.

4 Experiment

The reported experiment consists of two parts: facial expressions of the author and facial expressions of an elderly person. It aims to investigate the problems that occur in attempting to recognize facial expressions during conversations employing the Eigenface algorithm-based method by analyzing the experimental result.

4.1 Facial Expressions of the Author

For the first experiment, the training and testing dataset used the author's facial expressions. Five facial expressions are selected for the experiment: neutral, happy, sad, angry and surprise. For the training set we have obtained a range of images acquired in different contexts, i.e., in home and in work, with glasses and without glasses, under different background and different lighting conditions. Currently, only some images are used. In the first experiment, 23 images are in the training dataset and 9 images are in the testing dataset. The images used in the testing dataset are shown below in Fig. 2:

Fig. 2. Testing dataset

In addition, the facial expressions of the images in the training dataset are also included in a text file as the input. Results for the test of recognition of facial expression is shown in table 1 below. The first column is the number of image in the testing dataset. The second column is the actual facial expression displayed in each image. The third column is the facial expression recognized by the Matlab program. The Matlab program was developed by Tanveer and(which is described in [3]). The fourth column is the comparative results from the actual facial expression and the expression recognized by the Matlab program. As shown in Table 1, all the facial expressions in the 9 testing images are recognized correctly.

Table 1. The experiment result with author's own facial images

Test Image Number	Actual Facial Expression	Expression Recognized	Test Result
1	Neutral	Neutral	Correct
2	Neutral	Neutral	Correct
3	Happy	Happy	Correct
4	Sad	Sad	Correct
5	Sad	Sad	Correct
6	Angry	Angry	Correct
7	Angry	Angry	Correct
8	Angry	Angry	Correct
9	Surprise	Surprise	Correct

Fig. 2. 2nd testing dataset

In the second experiment the facial expressions employed are either positive emotions (happy, enjoyment, relaxed), negative emotions (unhappy, sad, angry) or emotionally neutral. This experiment investigated how to recognize the author's facial expressions when talking. There are 28 images in the testing dataset and 17 images in the training dataset. The images in the testing dataset are shown in the Fig. 3. The recognition accuracy of facial expression is 85.7%. In particular, images 5, 25, 26, & 28 are inaccurately classified. Naturally occurring emotional life facial expressions show more changes and variation. Moreover, there are changes in the head pose and mouths. Thus, the complexity of the task of recognition complexity is increased.

4.2 Facial Expressions of an Elderly Person

For the experiment employing the facial expressions of an elderly person, screenshots of a video of the interview of one elderly person were used as the training and testing dataset. There are 20 images in the testing dataset and 20 images in the training dataset. The images of the elderly person are not shown in the paper, because of some privacy reasons. This experiment was designed to test the Eigenface algorithm for facial expression recognition of elderly people in a natural communication situation.
Facial expressions in the images in the training dataset are also included in the text file as the input. Table 2 illustrates the result of recognizing the facial expression of the elderly person. The first column is the number of images in the test dataset. The second column is the actual facial expression displayed within each image. The third column is the facial expression recognized by the Matlab program. The fourth column is the comparative results from the actual facial expression and the expression recognized by the Matlab program. The result of the recognition of facial expression for the selected elderly person is 70.0%, with images 7, 8, 9, 10, 19, 20 were incorrectly classified. Again, we observed that in an accurate real-life situation, natural facial expressions show more changes and variation, including some changes in head pose and mouth shape. Thus. difficulty of recognition is greatly increased.

Table 2. The result for facial expression recognition

Test Image Number	Actual Facial Expression	Expression Recognized	Test Result
1	Happy	Happy	Correct
2	Sad	Sad	Correct
3	Sad	Sad	Correct
4	Happy	Happy	Correct
5	Neutral	Neutral	Correct
6	Neutral	Neutral	Correct
7	Sad	Neutral	Wrong
8	Sad	Neutral	Wrong
9	Sad	Neutral	Wrong
10	Sad	Neutral	Wrong
11	Sad	Sad	Correct
12	Sad	Sad	Correct
13	Neutral	Neutral	Correct
14	Neural	Neural	Correct
15	Happy	Happy	Correct
16	Neutral	Neutral	Correct
17	Neutral	Neutral	Correct
18	Neutral	Neutral	Correct
19	Happy	Neutral	Wrong
20	Happy	Neutral	Wrong

Of particular interest, we observed that a wry smile, which may be considered a negative feeling, is recognized as a happy emotion by the system. As a result, some sad emotions are recognized wrongly in the experiment. The problem mainly results from recognition of the facial expression by calculating the Euclidean distance to the nearest facial expression in the training dataset. The problems that observed in the process of facial expression recognition can be summarized as follows: firstly, that recognition of facial expressions by calculating similarity may result in incorrect recognition for some emotions, such as a wry smile. Secondly, during conversation, the lower parts of the facial features may change in various ways during speech. On the other hand, currently, the facial expression is recognized by finding the most similar image in the training dataset to the image in the testing dataset by calculating the Euclidean distance. As a result, the recognition result may be affected by some factor like appearance of the person. For future improvement, the facial expressions should be recognized using action units, or be classified in detailed categories. Additionally, the issue of changes to lower facial features during speech requires a solution. Thirdly, the performance of facial expression recognition needs to be compared with other algorithms in more detail. Also, dataset consisting of more images should be used. Finally, the influence of the key parameter in the PCA process should be discussed.

For extension of the work in the future, the computer vision techniques could be applied into other machine learning approaches and applications by further research. Relevant research work should be learned. For example, Jawad *et al* proposed a word-based off-line recognition system using Hidden Markov Models [16]. Jinchang *et al* proposed an improved neural classifier for early detection of breast cancer [17]. Yanmei *et al* researched about hierarchical and multi-featured fusion for effective gait recognition [18]. There are some other related tasks including human-computer interaction, unsupervised image saliency detection, handwritten Arabic Scripts Recognition and automatic gait recognition [19–22].

5 Conclusion

This paper described the process of recognizing facial expressions using the Eigenface algorithm, which was conducted using Matlab. First, this paper gave an introduction to the recognition of facial expressions. Next, the fundamental knowledge relating to PCA was provided; then, the Eigenface process in Matlab was detailed. In addition, an experiment on the result of experimental testing of the recognition of both the author's facial expression, and the expressions of one elderly person were reported.

In this approach, although the simplicity of comparing and classifying the facial expressions is an advantage, there are some drawbacks. For instance, this approach relies on a good training set. It cannot recognize facial expressions or facial features that are not included in the training set. In addition, when there are some inferences, the recognition accuracy will be affected. Moreover, problems that occurred in the process of the recognition of some less common facial emotions, those such as a wry smile, or that occur when lower facial features change during speech.

References

1. Machine learning for facial recognition – EFavDB, http://efavdb.com/machine-learning-for-facial-recognition-3/
2. Chen, J., Jenkins, W.K.: Facial recognition with PCA and machine learning methods. In: Midwest Symposium on Circuits and Systems. pp. 973–976 (2017)
3. Eigenface based Facial Expression Classification - File Exchange - MATLAB Central, https://uk.mathworks.com/matlabcentral/fileexchange/33325-eigenface-based-facial-expression-classification?requestedDomain=true
4. Fathallah, A., Abdi, L., Douik, A.: Facial expression recognition via deep learning. In: Proc. of IEEE/ACS Int. Conf. on Computer Systems and Applications, AICCSA. pp. 745–750 (2018)
5. Ou, J., Bai, X.-B., Pei, Y., Ma, L., Liu, W.: Automatic facial expression recognition using Gabor filter and expression analysis. In: ICCMS 2010 - 2010 Int. Conf. on Computer Modeling and Simulation. pp. 215–218 (2010)
6. Samal, A., Iyengar, P.A.: Automatic recognition and analysis of human faces and facial expressions: a survey. Pattern Recognit. 25, (1992). doi:10.1016/0031-3203(92)90007-6
7. Fasel, B., Luettin, J.: Automatic facial expression analysis: A survey. Pattern Recognit. 36, (2003). doi:10.1016/S0031-3203(02)00052-3
8. Sandbach, G., Zafeiriou, S., Pantic, M., Yin, L.: Static and dynamic 3D facial expression recognition: A comprehensive survey. Image Vis. Comput. 30, (2012). doi:10.1016/j.imavis.2012.06.005
9. Zeng, Z., Pantic, M., Roisman, G.I., Huang, T.S.: A survey of affect recognition methods: Audio, visual, and spontaneous expressions. IEEE Trans. Pattern Anal. Mach. Intell. 31, (2009). doi:10.1109/TPAMI.2008.52
10. Sariyanidi, E., Gunes, H., Cavallaro, A.: Automatic analysis of facial affect: A survey of registration, representation, and recognition. IEEE Trans. Pattern Anal. Mach. Intell. 37, (2015). doi:10.1109/TPAMI.2014.2366127
11. Turk, M., Pentland, A.: Eigenfaces for recognition. J. Cogn. Neurosci. 3, (1991)
12. Lo, H.-C., Churig, R.: Facial expression recognition approach for performance animation. In: Proc. - 2nd Int. Workshop on Digital and Computational Video, DCV 2001 (2001)
13. Wijeratne, S., Jayawardena, S., Jayasooriya, S., Lokupathirage, D., Patternot, M., Kodagoda, G.N.: Eigenface based automatic facial feature tagging. In: Proc. of the 2008 4th Int. Conf. on Information and Automation for Sustainability, ICIAFS 2008 (2008)
14. Toure, M.L., Beiji, Z.: Intelligent sensor for image control point of eigenfaces for face recognition. J. Comput. Sci. 6, (2010). doi:10.3844/jcssp.2010.484.491
15. Wu, C.-H., Wei, W.-L., Lin, J.-C., Lee, W.-Y.: Speaking effect removal on emotion recognition from facial expressions based on eigenface conversion. IEEE Trans. Multimed. 15, (2013). doi:10.1109/TMM.2013.2272917
16. Alkhateeb, J.H., Ren, J., Jiang, J., Al-Muhtaseb, H.: Offline handwritten Arabic cursive text recognition using Hidden Markov Models and re-ranking. Pattern Recognit. Lett. 32, 1081–1088 (2011). doi:10.1016/j.patrec.2011.02.006

17. Ren, J., Wang, D., Jiang, J.: Effective recognition of MCCs in mammograms using an improved neural classifier. Eng. Appl. Artif. Intell. 24, 638–645 (2011). doi:10.1016/j.engappai.2011.02.011
18. Chai, Y., Ren, J., Zhao, H., Li, Y., Ren, J., Murray, P.: Hierarchical and multi-featured fusion for effective gait recognition under variable scenarios. Pattern Anal. Appl. 19, 905–917 (2016). doi:10.1007/s10044-015-0471-5
19. Ren, J., Vlachos, T., Argyriou, V.: Immersive and perceptual human-computer interaction using computer vision techniques. In: 2010 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition - Workshops, CVPRW 2010. pp. 66–72 (2010)
20. Yan, Y., Ren, J., Sun, G., Zhao, H., Han, J., Li, X., Marshall, S., Zhan, J.: Unsupervised image saliency detection with Gestalt-laws guided optimization and visual attention based refinement. Pattern Recognit. 79, 65–78 (2018). doi:10.1016/j.patcog.2018.02.004
21. Alkhateeb, J.H., Ren, J., Jiang, J., Ipson, S.S., El Abed, H.: Word-based handwritten arabic scripts recognition using DCT features and neural network classifier. In: 2008 5th Int. Multi-Conf. on Systems, Signals and Devices, SSD'08 (2008)
22. Chai, Y., Ren, J., Zhao, R., Jia, J.: Automatic gait recognition using dynamic variance features. In: FGR 2006: Proc. of the 7th Int. Conf. on Automatic Face and Gesture Recognition. pp. 475–480 (2006)