Face Recognition with Deep Neural Network in Real-World Scenes

Tong Bing Ko[1, *], Yue Xi[2] and Yue Yin[3]

[1]Guangdong Experimental High School a Level Programme, Guangzhou, Guangdong, China

[2]No.3 High School, Harbin, Heilongjiang, China

[3]Ningbo Foreign Language School, Ningbo, Zhejiang, China

*Corresponding author: j.tong0426@icloud.com, xy510525xy@163.com, unicoursium@outlook.com

Abstract. The primary aim of this paper is to recognize faces using deep learning models. This method is employed in many areas such as human-computer interaction, automatic image indexing, and ID verification services. Despite its many uses, face recognition still has several difficult features, such as the head position, age, lighting, and facial emotions. In this paper, our method includes three components which are getting the datasets, training the face model, and recognizing faces. Three steps are included in our process. First, the authors introduce the Haar-like features and the first module-get the database by detecting the face, removing the background, and creating the grayscale images of face, then the authors introduce the second module by using LBPH method (Local Binary Patterns Histogram) to detect differences between the model sample face and the detected face. Last, the authors use face recognition predictor to return the recognition result and confidence. Finally, our method successfully recognizes people and show the confidence value at the same time. The experimental results show great potential of our method in face recognition with high accuracy.

Keywords: face recognition, Haar-like features, Local Binary Pattern Histogram.

1. Introduction

The goal of face recognition research is to compare a face in a digital image to a database of faces. This technique is used in ID verification services, automatic image indexing, and human-computer interaction. Despite its widespread applications, there are still some challenging aspects of face recognition including the head position, ages, light condition, and facial expressions.

Regarding real-time face detection and recognition in complicated backgrounds, see reference [1], the aim is to recognize the face even in a complex background. The algorithm is implemented using a series of signal processing methods including LBP, PCA, Haar-like feature, etc. To quickly recognize faces, facial features are extracted using the LBP descriptor. The false detection rate is decreased by the eye detection technique. High accuracy in facial recognition is maintained by processing the detected facial image to improve contrast and correct orientation. Effective face recognition is achieved using the PCA method. The methods obtain an overall true-positive rate of 98.8% for face detection and 99.2% for accurate facial identification by training the model with a vast collection of faces and non-faces images.

In studies on face recognition systems for blurred images, backpropagation neural networks and the method of extracting zoning information are used. The goal of this research is to find a solution to the issue of face recognition still having difficulties in abnormal photos that contain blur or noise and abnormal image analysis. Thresholding and zoning, known as FGTZ, were carried out using the BNN (Backpropagation Neural Networks) technique. In order to verify the efficacy of this technique, blur an image with changes of stances and lighting at levels 1 through 5 and 10. The outcomes demonstrated that an all-encompassing system with blur levels and poses may effectively discriminate between male and female. Additionally, the created system has been successful in recognizing faces with an average accuracy of over 79% in a variety of blur levels and positions.

The goal of research on supervised and unsupervised neural networks for 3D face recognition is to analyze the performance of two classifiers based on supervised and unsupervised learning. The

two techniques used for classification and regression analysis are Support-Vector Machines (SVM), which are supervised learning models with associated learning algorithms that study data and detect patterns. A form of artificial neural network (ANN) is trained using unsupervised learning to build a low-dimensional (usually two-dimensional), discretized representation of the input space of the training samples, known as a map. The basic SVM takes a collection of input data and predicts for each given input. Self-organizing maps are distinct from other artificial neural networks in that they employ a neighborhood function to preserve the topological characteristics of the input space belonging to the supervised and unsupervised approaches, respectively, and have been applied to two different 3D facial datasets. In order to determine which of the two primary ANN-based classifiers is more effective for a 3D face recognition system, the recognition rates have been assessed using the Linear Discriminant Analysis (LDA) approach. Two conclusions may be drawn from the collection of data generated by the tests performed using the LDA method: The first is that the unsupervised learning approach anticipates that we won't be aware of the ultimate result. According to the process's numerical results, a second conclusion can be drawn. SOM and SVM both performed well throughout the trial, although SOM's recognition rates in the test fall as the size of the LDA subspace does.

In our paper, OpenCV was used in this study. The methods utilized for this investigation is initially described in this paper. The research's findings are the subject of the third chapter. The findings analysis, potential future improvements, and potential study subjects are the three main elements in the fourth section. This research includes three components. The first module is to get the database by detecting the face, removing the background, and creating the grayscale images of face. The second module uses LBPH method (Local Binary Patterns Histogram) to detect differences between the model sample face and the detected face. The third module uses the function cv2.face FaceRecognizer.predict() to return the recognition result and confidence value.When our test person appeared in front of the camera, the system successfully recognized the person and gave the person's name. It then gave an accuracy of 82%.

2. Method

2.1. Haar-like Feature

For object recognition, digital picture features that approximate Haars are used. Their name comes from how intuitively they resemble Haar wavelets. They were used in the first module. A window the target size is moved across the input picture during the detection step. Haar-like feature is computed for every area of the image. Then, this distinction is put up against a threshold that is learned to discriminate between things and non-objects. The main advantage of a Haar-like feature is calculation speed. Integral images may be used to determine a Haar-like feature of any size in constant time (approximately 60 microprocessor instructions for a 2-rectangle feature).

2.2. LBP Feature

A technique called Local Binary Pattern (LBP) is used to illustrate how surfaces' textures are made up. LBP can be used to summarize texture pattern probability into a histogram. Each and every image pixel needs to have its LBP value determined. In addition to face and facial expression identification, the LBP has been used in a wide range of other biometric applications, including eye localization, gait recognition, palmprint recognition, iris recognition, and facial age classification.

2.3. Face Recognition Predictor

When people engage in social settings, the human face transmits significant social cues. In order to evaluate people's face-processing skills, the following variables were measured: recognition accuracy, reaction time, sensitivity, and response bias. There's an algorithm that allowed them to predict if someone would be good or bad looking based on their age and gender. By giving the A.I. photos and tell the A.I. the age and gender, it will start to learn. When it is ready, it can detect a new person age and gender. Some can even recognize the emotion of the person. Some companies has

used this algorithm for many years to help customers decide whether they should hire new employees, whether they should send out new proposals for projects, or whether they should reject customers who are not suitable for their business.

3. Experiment

3.1. First Program

For how face recognition is implemented, OpenCV is a ready-made method. The authors made 3 programs to achieve face recognition. In order to get the datasets the authors need, which are the images of the faces, The first program was created to do that. The first program need to detect the face, and remove the background and keep only the grayscale image of the face. A mouth that is darker than the background color, eyes that are darker than the cheeks, sides of the nose that are darker than the bridge, and other set components are included in each face's portrayal. Simply put, these traits can be categorized as rectangular Haar-like properties. The three categories of Haar-like features are edge features, linear features, center features, and diagonal features, as illustrated in Figure 1[4]. Furthermore, the Haar cascade classifier is built on a property shared with Haar. The two rectangular characteristics in Figure 2[4] below denote specific features of the human face. For instance, the right one reveals that the nasal bridge is lighter on the sides of the nose, while the center one suggests that the color of the eye area is darker than the color of the cheek area. Given little data, feature-based detection may encode region-specific states, and feature-based systems are significantly quicker than pixel-based systems, which is a benefit of the haar cascade classifier.

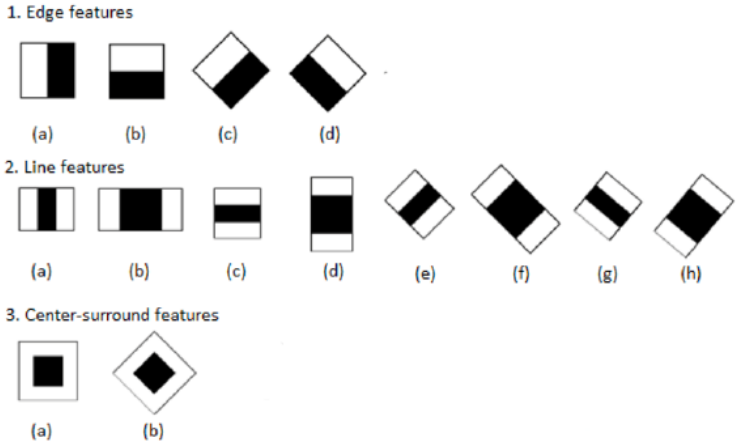

Figure 1. Three types of haar-like features

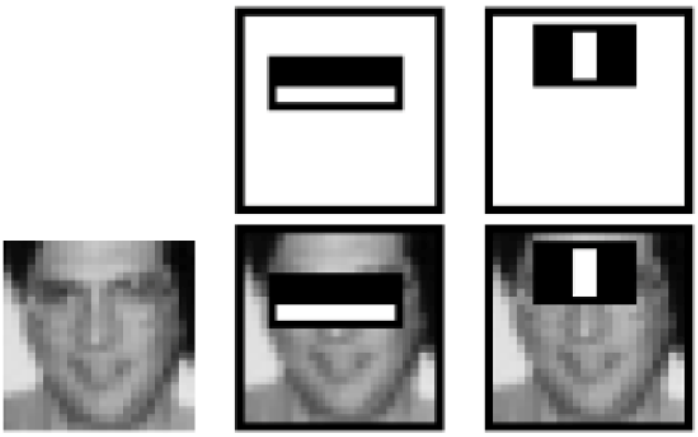

Figure 2. Example of the feature

3.2. Second Program

After the authors run the first program, the authors get the grayscale image of the target face. With a sufficient data set, it is time to run the second program to train the face model. OpenCV provides three face recognition methods, which are LBPH method, EigenFishfaces method and Fisherfaces method. In the program the authors use the LBPH method (Local Binary Patterns Histogram). The LBPH algorithm allows for differences in shape and size between the model sample face and the detected face, which is why the authors chose it. When the program is running, it will first take each pixel as the center, assess how it relates to surrounding pixels in terms of gray value, and binary encode it to obtain the LBP encoded image of the entire image (Figure 3[5]); after that, it will divide the LBP image into regions, obtain the LBP encoded histogram of each region, and then obtain the LBP encoded histogram of the entire image. Face recognition will then be achieved by comparing the LBP encoded histogram (Figure4). In order to recognize faces, LBP coding histograms of several face pictures are compared. To complete the second program, there are three phases. The first stage involves importing faces using the Python os module; the second involves training the faces using the LBPH method; and the third involves saving the learned model into a yml file so that a third application may match faces when recognizing them.

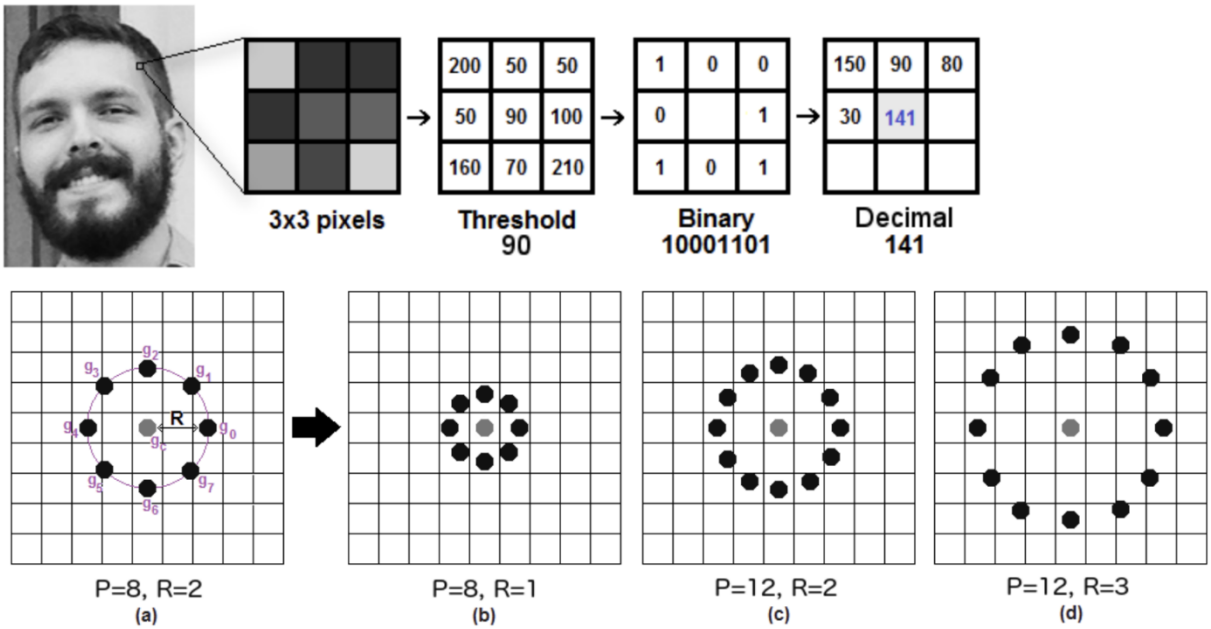

Figure 3. Determine the grayscale value of surround-ing pixels

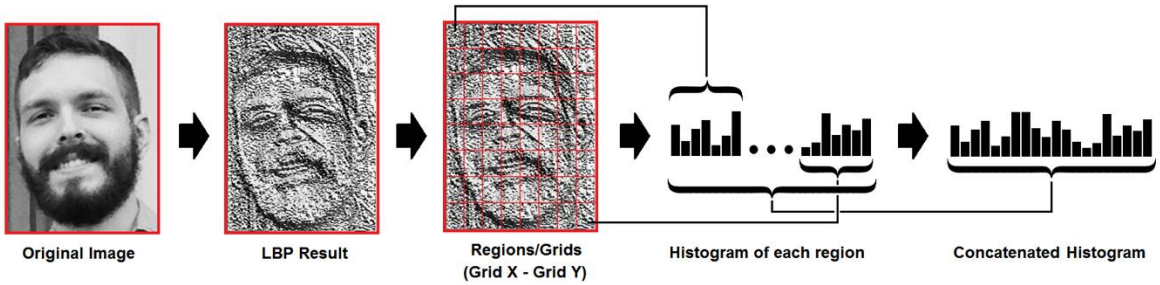

Figure 4. Generate LBP result and LBPH

3.3. Third Program

After training the AI model, the authors can run the program 3 when the authors want to recognize the face. The program 3 will take the photo of the person, and the code cv2.face FaceRecognizer.predict(src) in the program 3 will return the recognition result person and confidence value. This code will judge a face image to be measured and find the face image with the closest

distance to the current image. With which face image is closest, the current image to be tested is labeled with its corresponding label. If the distance between the current image to be tested and all face images is greater than the distance value specified by the parameter threshold in cv2.face.LBPHFaceRecognizer create(), then it means that the current face cannot be recognized. For example, when the authors enter the face data of two people into the model, and the labels of these two people are 0 and 1, then when the third program is run, the person with label 0 is photographed, then the program will output label=0 and the confidence value, and the same for the person with label 1. But when a stranger is photographed, the program will output that this is a stranger.

4. Results

When the authors run the program 1, the program will take 200 pictures of me, and then save them to the folder imgdata (Figure 5). And the authors can start the program 2 to train the images. When the images were trained, the authors can find a file called trainer.yml in the folder trainer, which is the AI model of me. And the authors can now recognize people. When I was taken photo, the name of me and the confidence value will show in the camera windows. (Figure 6). When a stranger are in the camera, it will show unknown.

Figure 5. Training example.

Figure 6. Inference demo with recognized and unknown faces.

OpenCV is a great method for us to recognize face, it can provide many ways for us to detect faces and train the faces into models. It is more accurate and can precisely distinguish between strangers and targets. And it supports real-time face recognition, capturing video streams with the camera and outputting confidence values, which reflects it's speed.

5. Conclusion

In our paper, the authors aim to introduce an algorithm for face recognition, which has three main modules. The first module is to get the database by detecting the face, removing the background, and creating the grayscale images of face. The second module uses LBPH method (Local Binary Patterns Histogram) to detect differences between the model sample face and the detected face. The third module uses the function cv2.face FaceRecognizer.predict( ) to return the recognition result and confidence value. The authors give better visualization results on real data, and the authors achieve 82% accuracy on the database.

Nowadays, facial recognition is integrated into both consumer applications like smartphones and many other fields in smart cities, such as object identification to identify objects that are out of place[6]. According to Juniper Research[7], the amount of hardware used for facial recognition increases by 50 percents annually. By 2024, 1.3 billion devices are anticipated to have facial recognition software. Companies like iProov and Mastercard are already using face recognition software in mobile phones that is powered by AI to authenticate payments and perform other sophisticated authentication activities. These applications will develop as the authors advance into 2022 and beyond. Researchers from University of New South Wales found that focusing on someone's ears and facial features increases accuracy by 6 percents.[8] This represents a considerable gain as even trained face identification personnel can compare photographs of unfamiliar faces incorrectly up to one out of every two times. Facial recognition is also being used by home security systems to increase access, facilitate security around the home, and offer a more seamless experience. Especially when deployed in smart home or building developments.

Systems for facial recognition contain flaws. Systems for facial recognition have come under fire for maintaining and making judgments based on a binary gender assumption[9]. While these techniques are typically extremely effective in identifying the gender of cisgender people's faces as either male or female, they frequently get it wrong or are unable to do so when it comes to determining the gender of transgender and non-binary persons. Even when given with a photo of a cisgender guy with long hair, these systems' commitment to supporting gender norms caused the algorithms to struggle between upholding the gender stereotype that men have short hair and the characteristics of males, leading them to become bewildered. This inadvertent misgendering of people may be quite damaging to those who do not identify with the sex assigned to them at birth by rejecting and discarding their gender identity. It is negative for those who don't follow conventional gender standards since it weakens people's gender expression regardless of their gender identification. Therefore, additional research on this subject may be necessary.

References

[1] Xin Zhang, Thomas Gonnot, Jafar Saniie. Real-Time Face Detection and Recognition in Complex Background. Journal of Signal Information Processing, 2017, 8, 99-112

[2] Wawan Setiawan. Face Recognition System for Blur Image Using Backpropagation Neural Networks Approach and Zoning Features Extraction Method. International Journal of Soft Computing 11 (1), 36-44, 2016

[3] Vaibhav M. Pathak, Suhuas S. Satonkar, Dr. Prakash B. Khanate. Analysis of Faces Using Supervised and Unsupervised Neural Network for 3D face Recognition. International Journal of Advanced Research in Computer Science and Management Studies, Volume 3, Issue 9, 2015

[4] Haar Samwell-Tarly-CSDN Haar. http://blog.csdn.net/yang6464158/article/details/25103703. [Online; accessed 2022-07-27].

[5] Lbph. http://panchuang.net/?p=21054. [Online; accessed 2022-07-29].

[6] What is the future of facial recognition technology in 2022 and be- yond? https://www.nec.co.nz/market-leadership/publications-media/what-is-the-future- of-facial-recognition-technology-in-2022-and-beyond/, dec 7 2021. [Online; accessed 2022-07-27].

[7] Facial Recognition Hardware to Feature on over 800m Mobiles by 2024. https://www.juniperresearch.com/press/facial-recognition-hardware-to-feature-on- over-800. [Online; accessed 2022-07-29].

[8] The Six-Minute Training Hack That Can Improve Face Recognition Skills. https://neurosciencenews.com/facial-recognition-hack-19816/, dec 19 2021. [Online; ac- cessed 2022-07-28].

[9] Contributors to Wikimedia projects. Facial recognition system. https://en.wikipedia.org/wiki/Facial recognition systemIneffectiveness, jun 27 2022. [Online; accessed 2022-07-29].
[19] A short-term network load forecasting model and method based on neural network and fuzzy logic. China journal, 2004, 19(10)：53-58. Harbin University of technology, 2011.

[10] To predict short-term hourly load, amjadi n. IEEE Journal of electrical systems, 2001, 16 (4): 798-805.

[11] McCullen. Short term load distribution prediction based on big data. Changsha: Hunan University, 2014.

[12] Shi Biao, Li Yuxia, Yu Xiaohua, Wang Yang: short term load forecasting based on improved particle swarm optimization algorithm and fuzzy network model. Theory and practice of system construction, 2010, 30 (1): 158-160.