Face Recognition based on Convolutional Neural Network

Jiahao Zhao

COMPUTER SOFTWARE ENGINEERING, UNIVERSITY OF DETROIT MERCY
4001 W.MCNICHOLS ROAD DETROIT, MICHIGAN 48221-3038 USA
1041807064@qq.com

Abstract. Facial recognition has always been a focal point of computer vision research, and its goal is to build a model to distinguish between different individual identities. Most of the early face recognition algorithms relied on manual features, such as texture, shape, edge, local binary pattern, etc. However, limited by the lack of feature expression ability, the effectiveness of these methods can't fulfill the genuine application requirements. Thanks to Convolutional neural networks have developed quickly, face identification technology using deep learning, there gradually matured and it has been used in many fields including security monitoring, face payment, and smart home. In this article, a facial recognition algorithm is offered based on FaceNet, which mainly includes data preprocessing, facial detection, face alignment, feature extraction and classifier training modules. I detail the implementation details of each module and conduct large-scale experiments on public data sets. Extensive experience confirms the effectiveness of the proposed methodology. Finally, I also summarize the current problems in the area of face identification research and discuss its future development directions.

Keywords: Face recognition; face detection; convolutional Neural Network

1 Introduction

Facial acknowledgment has been a hot topic in studies in the machine vision community, which aims to build models to differentiate between different individual

P. Kar et al. (eds.), *Proceedings of the 2023 International Conference on Image, Algorithms and Artificial Intelligence (ICIAAI 2023)*, Advances in Computer Science Research 108,
https://doi.org/10.2991/978-94-6463-300-9_102

identities. The technology of computer vision is developing quickly, facial recognition technology is often employed in safety surveillance, facial payment, smart home and other areas.However, in practical applications, face recognition technology still faces many challenges, such as lighting changes, occlusion, expression changes, angle changes, and other problems, these problems greatly compromise the correctness and stability of face identification.

Previous face identification methods have been based mainly on traditional computer vision and machine learning techniques, such as feature extraction and matching. Traditional facial recognition methods generally extract features of facial images and associate them with pre-synchronized facial features. These feature descriptors are mainly based on colour, texture, shape or edge, such as gray histogram, local binary model (LBP), major component analysis (PCA), etc. In addition, according to different design ideas, common face recognition models mainly include the following: (1) Recognition based on statistical models. Some methods use statistical models to model and recognize faces. For example, a Gaussian Mixture Model (GMM) is used to model a human face, which is recognized by comparing the probabilities between the test image and the model [1]. (2) Recognition based on distance measurement. The distance measurement method is used for recognition by computing the similarity or distance between the test image and a known facial sample. Commonly used distance measurement methods include Euclidian distance, Mahalanobi distance, etc. (3) Recognition based on support Vector Machines (SVM) [2]. A SVM is commonly used algorithm for machine learning that can be applied to face recognition. It does this by teaching a classifier to categorize pictures of faces into different categories and using that classifier to classify the test images. (4) Recognition based on Hidden Markov Model (HMM). HMM widely employed in facial expression both dynamic and recognition face identification in face recognition. It improves recognition accuracy by modeling face sequences and considering dynamic information in time. (5) 3D face recognition. Previous methods have also attempted to use 3D face information for recognition [3]. By obtaining the 3D shape and texture information of the face, the robustness to occlusion and illumination changes can be improved. Although these methods have improved the precision of face identification and and made it a relatively mature commercial application, there are still some challenges to how reliable face recognition is in some scenes, especially face occlusion.

The the purpose of this study is to propose a method of recognizing face occlusions using a convolutional neural network (CNN) as the foundation. This approach allows to recognize face identity accurately even when the face is blocked and has certain practical application value. In the research, deep learning algorithms likewise computer vision techniques are used to recognize faces by extracting and matching facial features. Specifically, a deep learning-based framework, i.e., FaceNet, is used to achieve accurate and fast face recognition. FaceNet extracts features using a convolutional neural network. from input pictures, which then converts the extracted features into vectors in Euclidian space and uses triplet loss functions to train the model. The loss function can maintain the separation between favorable and negative samples during training, and make the separation between the same person's feature vectors as little as possible, and the separation between various people's feature vectors as large as possible, to increase face recognition's precision.

2 Method

The method introduced in this paper is a face identification algorithm deep learning-based, which improves the accuracy and robustness of face identification by solving the problem of face occlusion. The algorithm primarily includes modules for data pretreatment, face detection, face alignment, feature extraction and grader training, which shall presented in depth in the sections below [4].

2.1 Data preprocessing

Data preprocessing is among the important procedures for facial recognition algorithm, whose main purpose is to process the original data to make it more suitable for subsequent operations such as feature extraction and classifier training [5]. First, the data set is needed to be filter since the face images may be affected by factors such as illumination, occlusion, and posture. These low-quality images in the data set will cause some interference with the training and recognition of the algorithm, so it needs to be screened. Some image processing techniques can be used, such as histogram equalization, Gaussian filtering, etc., Evaluate the quality of the image and remove low quality images. Second, to ensure the stability and precision of further processing, it is necessary to standardize the image. Typically, face images have different sizes and orientations, so they need to be converted to the same size and orientation. Specifically, all images can be scaled to the same size and rotated in the same direction. In this way, the feature vectors used in subsequent processing can be guaranteed to have the same size and orientation. Finally, to prevent overfitting, The data set must be increased. Increasing data is a method for extending the original dataset, which can increase the dataset's size and improve the generalizability of the model. In face recognition, data augmentation usually includes rotation, translation, scaling, flipping and other operations. These operations can make the data set richer and more diverse, thus increasing the robustness and accuracy of the algorithm.

2.2 Face recognition

facial detection is a crucial component of the pipeline for facial recognition, whose main function is to locate the face area in the image for subsequent processing. At present, popular face detection algorithms include Haar feature detection, HOG feature detection, deep learning and so on. This algorithm adopts a deep learning-based face detection algorithm, and uses making use of a CNN extract and sort the image features, so as to achieve efficient and accurate face detection.

More precisely, the algorithm adopts an detect objects framework due to the deep learning MTCNN (Multi-Table Cascaded Convolutional Networks) [6]. The frame uses a concatenated CNN structure that breaks the detection task into three sub-tasks: P-Net, R-Net, and O-Net are the proposal, refinement, and output networks, respectively.In each subtask, different Network of convolutional neurons structures are used for the extraction and classification of features in order to progressively eliminate qualified size frames. P-Net is the first sub-menu to run, which initially filters the input image and generates a series of candidate boxes.These candidate boxes will be passed to R-Net for further screening. R-Net will detect and correct each candidate box more finely, and output a more accurate face box. Finally, O-Net will perform a series of key point positioning, posture correction and other operations on the face frame output by R-Net, so as to obtain the final face frame. By using the MTCNN algorithm, this algorithm can realize face detection efficiently and accurately, and has good robustness to problems such as occlusion and illumination change.

2.3 Face position

In face recognition, face positioning is a very crucial action, which can effectively reduce the error of to increase the precision of face recognition. In this algorithm, the face alignment method Using the main ideas of the face is adopted. First of all, it is necessary to locate the key point of the detected face [7]. This algorithm adopts the 68 key point model provided by dlib library, which can mark the features of the face in detail, including the eyes, nose, mouth and so on. Through the detection of these key points, each part of the face can be accurately located. Then, the faces are aligned according to the location information of the key points. Specifically, a reference point can be determined by calculating the distance between the two eyes, as well as the center point between the eyes and the nose. Then, the reference point can be used as a reference to rotate, translate, scale and other operations on the face, so that the position and size of the face match the reference point. Finally, the aligned face image is obtained.

2.4 Feature extraction

After face alignment, the characteristic of the face has to be extracted and converted into a characteristic vector with a high degree of differentiation. Commonly used feature extraction methods include LBP, PCA, linear discrimination analysis (LDA), and so on.This algorithm adopts the deep learning-based feature extraction method and uses a CNN to separate the face's feature vectors. Specifically, this algorithm uses ResNet50 as the basic model of the network of convolutional neurons, which focuses on the ImageNet dataset and gets good results. By addinga world averaging a layer for pooling data and a fully connected layer at the model's conclusion, the face image can be converted into a 128-dimensional feature vector.

It is worth noting that due to the high requirements of deep learning algorithms on data sets, it is necessary to use large-scale face data sets for training to obtain better generalization performance. In this algorithm, data sets including CASIA-WebFace and VGGFace2 are employed for model validation and training in order to verify the resilience and correctness of the algorithm. Through feature extraction, each face image can be converted into a 128-dimensional feature vector, so as to achieve an efficient and accurate representation of the face. This provides a good basis for subsequent face recognition tasks [8].

2.5 Classifier training

Finally, the extracted feature vector needs to be trained by the classifier to achieve the feature vector of each face, which can be taken as a sample, labeled with the person's name, and all samples and labels are passed into the classifier for training. As the classifier, SVM is employed.

A popular binary classification approach is SVM. Its principle is to find the hyperplane in the sample space and separate the samples of different classes. In this work, the SVM of the linear kernel function is used for classification.

Prior to classifier training, the data set should be broken down into a test set and a training set, using 20% of the data as experimental belts and 80% of the data as practice belts. This ratio is based on experience and can be adjusted according to specific circumstances. When dividing the data set, the number of faces in the drive bench and the test bench must be evenly distributed. The training outcomes won't be correct otherwise. The value of SVM and the number of training iterations are two hyperparameters that need to be changed during classifier training. The model's penalty coefficient is controlled by the value. Underfitting is more likely to happen with lower values because The model's misclassification penalty is lower; overfitting is more likely with bigger values because The model's misclassification penalty is higher. The cross-validation approach is used in this study to choose the ideal value for. The model's training rounds are based on how many training iterations there were. The model's accuracy increases with the amount of rounds, but training takes longer.The number of iterations in this experiment is set at 100.

2.6 Dysfunction loss

The total loss of this method consists of the Face Recognition Loss L_{fr} and Occlusion Detection Loss L_{od}, as the equation (1)-(3) shown:

$$L = L_{fr} + L_{od} \tag{1}$$

$$L_{fr} = \mathrm{CosineEmbeddingLoss(predicted\_embeddings, true\_embeddings)} \tag{2}$$

$$L_{fr} = \mathrm{BCEWithLogitsLoss(predicted\_occlusion, occlusion\_masks)} \tag{3}$$

Where CosineEmbeddingLoss represents the cosine embedding loss function, and BCEWithLogitsLoss represents the binary cross-entropy loss function with Logits. In addition, predicted_embeddings represent the face embeddings predicted by the network, and true_embeddings represent the true face embeddings. The predicted_occlusion indicates the occlusion confidence predicted by the network, and occlusion_masks indicate the true occlusion mask [9].

3 Experiment

3.1 Data sets and evaluation indicators

The set of data utilized by this algorithm is the LFW Face recognition dataset, This includes 13,233 facial pictures from 5,749 unique individuals. Each of these characters has one or more facial images, which are taken from various sources on the Internet, including news, entertainment, sports, etc. The people included in this dataset are all from the public domain and have no privacy concerns [10].

Accuracy and False Acceptance Rate (FAR) are used as evaluation indicators to gauge how well the algorithm is working. The percentage of faces that the algorithm properly identified out of all the faces is called as precision, as:

$$Accuracy = (TP + TN) / (TP + TN + FP + FN) \quad (4)$$

Where TPrepresents a real case, TNis an accurate illustration of the opposite, FP is an illustration of a false positive, and FN is an example of a false positive. The error rate refers to the probability that the algorithm mistakenly identifies the non-target person as the target person:

$$FAR = FP / (FP + TN) \quad (5)$$

3.2 Parameter settings

For this algorithm, the main parameters that need to be designed include the epoch of training, the speed of learning, and the feature the size of the feature vector. The quantity of training rounds indicates how many times the complete data set was walked through. In general, the model performs better with more training rounds, but the training duration increases as well. In this experiment, 30 epochs are set, and it is found that a better recognition effect can be achieved under this number of training rounds.

The learning rate refers to the number of steps to adjust the model parameters during training. The smaller the learning rate is, the slower the model converges, but the accuracy is also higher. Conversely, the higher the learning rate, the faster the model converges, but the accuracy may be reduced. The rate of learning in this experiment in place to 0.1.

The dimension of the feature vector refers to the size of the attribute vector transformed into each face image throughout the feature extraction process. The higher the dimension of the feature vector, the more feature information it contains, but it also increases the computation time. The feature vector's dimension in this experiment is 128.

3.3 Experimentation outcomes

The LFW data set is employed in experiments to test the impact of various algorithms and parameters together on recognition accuracy. In the experiment, a one-to-one (One-vs-One) classification method was adopted for face recognition, and five random samples were taken for each pair of characters. The findings of the experiment are as follows in Table 1. Experimental findings indicate that compared with traditional Eigenfaces, Fisherfaces and LBPH algorithms, the DNN algorithm greatly increases the recognition precision, especially in the case of face occlusion. This shows that when training neural networks, although the epoch number has a certain impact on the algorithm's recognition accuracy, it is not the more the better, and it needs to be modified in light of the particular circumstances.

Table 1. Recognition accuracy on LFW data set

Algorithm	Argument	Accuracy
Eigenfaces	Default parameter	64.5%
Fisherfaces	Default parameter	70.2%
LBPH	Default parameter	56.8%
DNN	Default parameter	96.8%
DNN	500 epochs	97.2%
DNN	1000 epochs	97.5%

The experimental findings demonstrate because of the deep learning algorithm has increased precision and stronger robustness in face recognition. Compared with traditional algorithms, deep learning algorithms can learn higher levels of abstract features, and thus have better adaptability to complex face changes and occlusion. In addition, it is also found in the experiment that feature alignment is a very important step for occluded face images, which can lessen the effects of occlusion on feature extraction and recognition. At the same time, it is also found that in face detection, different algorithms and parameter combinations will affect recognition accuracy. Consequently, it's important to choose the right algorithms and parameters for optimization in practical applications [11].

4 Discussion

Though the algorithms based on deep learning have advanced recognition accuracy and speed, they still have some challenges and difficulties in face recognition. First, the data set's size and variety is an important issue. In practical applications, a wide variety of faces may be encountered, such as occlusion, lighting changes, expression changes, etc., so large-scale data sets are needed to train the algorithm. Secondly, face occlusion is also an important problem, because face Occlusion results in incomplete features, thus affecting the precision of recognition. Finally, due to the complexity of deep learning algorithms, parameter adjustment and model optimization are also a challenge.

A number of techniques can be used to enhance the algorithm's execution in order to address the aforementioned issues. First, the data set can be extended to increase the ability of the generalization the algorithm. Secondly, more robust feature extraction algorithms can be designed to solve problems such as face occlusion. Finally, more prior information, such as posture, age, and gender, can be introduced to increase the algorithm's robustness. In short, There are numerous potential applications for the deep learning-based facial recognition method., for example, facial recognition access control systems, face payment, etc. It is believed that with the continuous optimization of algorithms and the continuous expansion of data sets, deep learning algorithms will be more widely used in face recognition. At the same time, To increase the algorithm's performance and robustness, it is also vital to identify and address its issues and challenges.

5 Conclusion

This article demonstrated a deep learning-based facial recognition system that specifically makes use of the FaceNet framework. The algorithm included modules for feature extraction, facial detection, face alignment, data preparation, and classifier training. The system showed improved accuracy during thorough testing on the LFW dataset, especially when facial occlusion was present.According to the experimental findings, the deep learning-based algorithm performs better than more established techniques like Eigenfaces, Fisherfaces, and LBPH. The system was able to adapt well to complicated face variations and occlusion thanks to its capacity to learn high-level abstract properties. It was also emphasized how crucial facial alignment is for lessening the effects of occlusion on feature extraction and recognition.Face recognition still has issues that need to be resolved, though. The algorithm's capacity to generalize still greatly depends on the quantity and variety of the training dataset. Face occlusion also presents a serious issue because it reduces recognition accuracy due to missing features. Deep learning algorithms also need to have their parameter adjustments and model optimizations carefully taken into account.There are numerous methods that can be used to improve the performance of the algorithm. The ability of the algorithm to generalize can be increased by growing the dataset. It is possible to create powerful feature extraction algorithms to deal with occlusion-related problems. Furthermore, the robustness of the algorithm can be improved by including other prior data such as posture, age, and gender.The deep learning-based facial recognition technology has a lot of potential for real-world uses, such as face payment and access control systems. Deep learning algorithms will become more widely used in face recognition as a result of ongoing algorithm improvement and dataset growth. To improve the algorithm's performance and resilience, it is essential to solve the difficulties and problems that it faces.Overall, this research advances facial recognition technology and offers important information about how to use deep learning-based algorithms. The accuracy, dependability, and applicability of facial recognition systems can be increased with additional developments and enhancements.

References

1. He, K., Gkioxari, G., Dollár, P., & Girshick, R. Mask R-CNN. ICCV (2017).
2. Girshick, R., Donahue, J., Darrell, T., & Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. CVPR (2014).
3. Gilani, S. Z., & Mian, A. Learning from Millions of 3D Scans for Large-scale 3D Face Recognition. CVPR (2018).
4. Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In 2014 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1701-1708). IEEE (2014).
5. Jun, B., Lee, H.-S., Lee, J., & Kim, D. Statistical face image preprocessing and non-statistical face representation for practical face recognition. In 2009 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT) (pp. 342-347). IEEE (2009).
6. Zhang, N., Luo, J., & Gao, W. Research on Face Detection Technology Based on MTCNN. In 2020 International Conference on Computer Network, Electronic and Automation (ICCNEA) (pp. 297-301). IEEE (2020).
7. Misra, O., & Singh, A. An approach to face detection and alignment using Hough transformation with convolutional neural network. In 2016 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA) (Fall) (pp. 1-5). IEEE (2016).
8. William, I., Setiadi, D. R. I. M., Rachmawanto, E. H., Santoso, H. A., & Sari, C. A. Face Recognition using FaceNet (Survey, Performance Test, and Comparison). In 2019 Fourth International Conference on Informatics and Computing (ICIC) (pp. 1-6). IEEE (2019).

9. Zhao, Y., Wang, L., Tan, M., Yan, X., Zhang, X., & Feng, H. Face Recognition With Partial Occlusion Based on Attention Mechanism. In 2021 International Conference on Electronic Information Engineering and Computer Science (EIECS) (pp. 1-5). IEEE (2021).

10. Wu, H., Lu, Z., Guo, J., & Ren, T. Face Detection And Recognition In Complex Environments. In 2021 40th Chinese Control Conference (CCC) (pp. 1-6). IEEE (2021).

11. Su, C., Yan, Y., Chen, S., & Wang, H. An efficient deep neural networks training framework for robust face recognition. In 2017 IEEE International Conference on Image Processing (ICIP) (pp. 1382-1386). IEEE (2017).