Attention Mechanism Based Convolutional Neural Network for Unbiased Facial Recognition

Haoyang Zou[1], Jianbo Chen[2,*]

[1] Saint Anthony's High School, Ny 10001, usa

[2] University of Chinese Academy of Sciences, Beijing 100049, China

*Corresponding Author

Keywords: Attention mechanism, Neural network, Unbiased face recognition

Abstract: Facial recognition is an important and topical task in the field of artificial intelligence. In the last decade, a lot of algorithms have been developed to solve facial recognition (FR) problem. However, the prevailing facial recognition datasets are unbalanced in terms of age, gender, and race. Therefore, most of the previous algorithms trained on those datasets are biased. What's more, most of the previous methods lack the ability to efficiently focus on the salient regions of a facial image. To solve the above-mentioned problems, a much more balanced dataset containing people from different ages, races and genders are collected, and an attention based method is proposed to automatically focus on the most important and salient local regions. Extensive experiment has been conducted, and the results demonstrate the effectiveness of the proposed method.

1. Introduction

Facial recognition is a task to identify or verify people from their facial images. Facial recognition has been widely used in human daily life, such as cellphone unlock [1], security monitoring. [2], mobile payment [3], and so on [4,5,6]. Recently, lots of FR algorithms have been proposed in the literature, most of them can be classified into 3 types, i.e., handcrafted feature-based methods, sparse representation-based methods, and deep learning methods.

Traditionally, face recognition problem is solved with handcrafted feature [3,4,5,6,10]. For example, [3] use LBP-Based Face Descriptor to recognize faces. They divided face image into several regions from which the LBP feature distributions were extracted and concatenated as a face descriptor. It is one of the best performing texture descriptors and it has been widely used in various applications. However, this kind of methods is LBP only recognize shallow features of the face, it is still less effective for complex recognition problems of face features that are difficult to classify.

There are also approaches using Sparse Representation to preform face classification [4,8,9,10]. For example, J. Wright et al. [4] created a new Loss called A-Softness Loss, it learns separable features that are not discriminative enough and it defines a large angular margin learning task with adjustable difficulty. But the flaw of this method is the obtained sparse code may be highly redundant.

With the huge improvement of artificial intelligence, deep learning has becoming predominating in the field of face recognition in the last decade [5,11,12,13,14]. For example, [5] present a system that has closed the majority of the remaining gap in the most popular benchmark in unconstrained face recognition and is now at the brink of human level accuracy called Deepface. But this method had a big flaw, a fixed convolution kernel is used, and there is no weight sharing, which leads to excessive parameters and prone to over-fitting problems.

Although huge progress has been achieved, previous methods are still biased upon human ages, races, etc. What's more, most of those previous methods are not able to efficiently concentrate on the salient regions of the input face images. In contrast, we proposed an attention based deep learning method to automatically focus on the useful regions of the face to enhance the signal noise ratio. We use backbone networks like VGG [15], ResNet [16] & LeNet [17] to develop our algorithm based on these methods. With these methods, we can recognize specific human face from complex

Copyright © (2020) Francis Academic Press, UK DOI: 10.25236/mmmce.2020.023

background better.

In addition to the design of algorithm, face datasets also play a very important role in the performance of facial recognition. As facial recognition is attracting more and more attention in the computer vision community, a lot of face datasets has been collected and released [18,19,20,21,22]. However, most of the previous released datasets failed to take into consideration the data balance along different ages, races, and genders. Algorithms trained on these datasets are therefore tend to be biased and produce unfair outcomes. To relief the above mentioned problem, this paper collected a much fair dataset along different ages, ethnicities, and sexes. Algorithms trained on this dataset would be less biased and much fairer.

Extensive experiments are conducted in this paper to evaluate the performance of the designed algorithm on the collected balanced dataset. The results show that our proposed method is able to achieve the state of the art performance, which demonstrates the effectiveness of the method designed in this paper.

The rest of the paper is organized as follows. Section 2 mainly discusses the collected dataset. Section 3 elaborates our method in detail. In section 4, the experimental results of our method are discussed in depth. And finally, this work is concluded in section 5.

2. Dataset

This section goes into details about the collection process of our dataset, and the statistic metadata of the collected dataset.

In order to collect a dataset that is balanced and unbiased, we set some keywords that related to our dataset, especially for different races, ages and genders. Then, google is used as search engine for image retrieval. Google has the world-leading searching system, it contains large number of different samples we need. In addition to the variance of ethnicity, age, and gender, those data retrieved from internet also varies from facial poses, light conditions and face expressions. After image collection, the data are annotated with ground truth information. Labeling is a very time-consuming job, it took a long time to label each sample, for example, each image is labeled with the name, age, race gender of the subject.

In our collected balanced face dataset, there are 20 different identities, and each people have 100 different images. Therefore, there are 2k images in total in our database. What's more, there are different variables in 20 samples, for race, database include Asian, White, African American and Latino. For age, our database almost includes every age-stage, 15-30, 31-50, 50-70 and 70+. Compared with other database, they do not have a comprehensive database that include with these variables, for example, different brightness of light, different light reflection on human face, different resolution of each images.

Figure 1 10 different samples in the proposed database. Figure 1 includes samples with different variables. For races as Latino, Asian, White, Black; for ages, there are 20-25, 40-50, 70+. There are also different light conditions in each picture and each picture has different resolution.

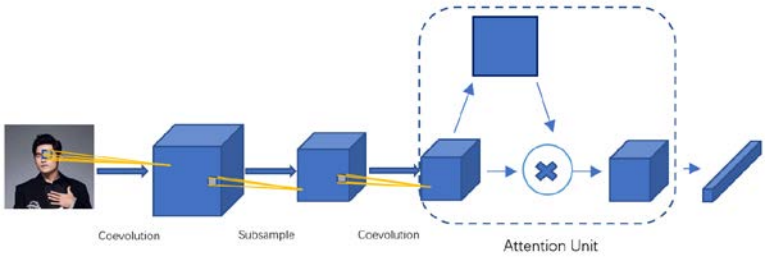

Fig.2 The Structure of Our Network Framework Diagram. the Feature Extracted from Neural Network is Further Processed in the Attention Unit to Filter out the Noise.

3. Method

This section elaborates the methodology of our work. Firstly, a basic convolutional neural network is implemented as the backbone of our algorithm, which is mainly depicted in subsection 3.1. Secondly, in order to force the network better focusing on the saliency area of the input image, an attention module is designed upon the CNN backbone, which is discussed in subsection 3.2. Finally, a cross entropy loss function is implemented for network training and elaborated in subsection 3.3.

3.1 Network Backbone

In the proposed algorithm, input facial images are supposed to be fed into a feature extractor to get an abstract representation. In general, the feature extractor could be any kind of neural networks or handcrafted feature extractors. Specifically, this paper mainly discuss the situation that neural network is used as feature extractor. The following parts uses LeNet as example. The receptive field of a neuron refers to a specific area, and only stimulation in this area can activate the neuron. When we load an image, for example, the size of the image is 32*32=1024, the image is our input. Then we convolute input and we will get a convolution layer, we call it C1 layer, the size of the filter is 5×5 = 25, and there are 6 filters in total. Obtain 6 sets of feature maps, the size is 28×28 = 784. Therefore, the number of neurons in the C1 layer is 6×784 = 4,704. The number of trainable parameters is 6 × 25 + 6 = 156. The number of connections is 156 × 784 = 122,304. Next, we subsample C1, we called the new layer we get S2 layer, the Sub-sampling layer. The 2×2 adjacent points in each group of feature maps in the C1 layer are sampled as 1 point, which is the average of 4 numbers. The number of neurons in this layer is 14×14 = 196. The number of trainable parameters is 6× (1 + 1) =12. The number of connections is 6×196× (4 +1) = 122,304. Then, we convolute again, we call this convolution layer “C3 layer”, due to the S2 layer also has multiple sets of feature maps, a connection table is needed to define the dependency between the feature maps of different layers. There are 60 filters in this layer, and the size is 5*5=25. Obtain 16 sets feature maps with a size of 10×10 = 100. The number of neurons in layer C3 is 16×100=1600. The number of trainable parameters is 60×25 + 16 = 1,516. The number of connections is 1,516×100 =151,600. Then, we subsample again, and get a new down-pooling layer called S4. Sampling from 2×2 neighborhood points into 1 point, 16 groups of 5×5 feature maps are obtained. The number of trainable parameters is 16×2 = 32. The number of connections is 16× (4 +1) = 80. Then, we convolute again, and get a new convolute layer called C5 layer. Obtain 120 groups of feature maps with a size of 1×1. Each feature map is connected to all feature maps in the S4 layer. There are 120×16 = 1,920 filters, the size is 5×5= 25. The number of neurons in the C5 layer is 120, and the number of trainable parameters is 1,920×25+120=48,120. The number of connections is 120× (16×25+1) = 48,120. Then, we move to F6 layer which is a full connection layer. There are 84 neurons, the number of trainable parameters is 84× (120+1) = 10,164. The number of connections is the same as the number of trainable parameters, which is 10,164. Finally, we can get our output layer.

In addition to backbone, our algorithm is more comprehensive than other algorithms because more appropriate processing methods have been applied to the face image before input to the network. We improved our algorithm by developing more targeted algorithms to make our algorithm

more precise. Other algorithms mostly have some same flaws with difficult to distinguish human face because of different angles and different light reflection. We collect 100 images for each person, each image is in different size, we develop an algorithm that can make every image to same size. The algorithm can shrink large size picture to a small size picture, and also make small size image bigger. We also develop an algorithm that specially for different brightness of light in the picture. Some light in the picture may influence the previous algorithm to distinguish human face from the background. We love this question by adjust light in the image. We adjust the exposure and brightness of the light in the picture, we make different exposure level and bright level. Then, we use our algorithm to analyze a new image with the best poses and brightness. To the point of different angles or poses, we develop an algorithm to solve this problem. After we finish previous procedures, we will get a better light of the picture, we use our algorithm to find human face. But sometimes there may only appear half side of face of people and it will be hard to determine if this part of the face belongs to the person we need to find. Our algorithm will extract key information of face and make turn the face into a front angle. After all these procedures, we will get a new 100 same size, same resolution with better exposure and brightness images and easy to distinguish human face from the background.

3.2 Attention Unit

Inspired by the human visual attention mechanism, an attention unit is implemented upon our model, which is illustrated in Fig 3. The attention unit is an important part of our algorithm. It is a system that could extract key information from complex images. For example, Kobe Bryant has unique nose and eyebrows, but in some cases, it may be difficult to distinguish this kind of key information from some noisy background, or some terrible light condition. To address with this problem, the attention unit can extract this kind of key information from complex images. Then, with that key information, we can match them with our database much more easily.

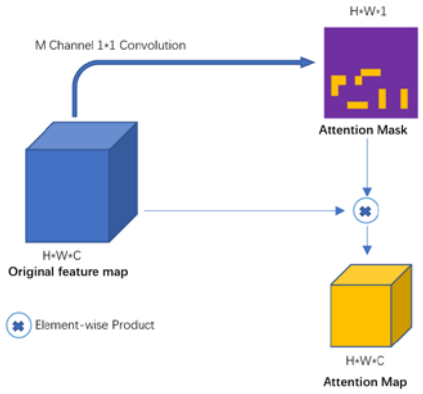

Fig.3 The Attention Unit in Our Algorithm. the Original Feature Map is Multiplied with the Attention Mask to Produce the Final Attention Map.

3.3 Loss Function

In addition to the design of network architecture, a proper loss function is also important to the task of face recognition. In this work, the cross-entropy loss is implemented to correctly classify the faces into the corresponding categories.

In machine learning algorithm, we usually define a loss function to determine if the decision function is good or bad. In our algorithm, we use cross entropy loss function. The entire loss function is formulated as Eq. 1:

$$L(y, f(x,\theta)) = -\sum_{i=1}^{C} y_i \log f_i(x,\theta), \quad (1)$$

where y is the label of the input image, f(x, θ) is the predicted value. C is the number of class. For cross entropy loss, the model output f (x, θ) is the conditional probability of each class y. yi is the distribution of the marked real categories, and the above formula happens to be the form of cross entropy.

4. Experiment

In previous sections, we already discussed about our method, algorithm, dataset and flaws about other methods. This section will discuss the experiment of the proposed model to evaluate its effectiveness. 4.1. Discuss about our experiment protocol. 4.2. Discuss about the experiment result.

4.1 Experiment Protocol

The proposed algorithm is mainly programmed with MATLAB. The dataset is randomly separated into training set and test set by the ratio of 8: 2. During the training procedure, the learning rate is set to 0.001 experimentally. The batch size is set as 64, and the number of epoch is 50.

Before feed into the proposed network, facial images are preprocessed with several procedure. Firstly, the face detection and face alignment are conducted with MTCNN []. Secondly, the face are cropped from the entire image to filter out the noisy background. Finally, data augmentation is implemented with random noise and horizontal flip.

Considering that the collected dataset in this paper only contains 2k images, directly using the data to train the network would result in severe overfitting. To avoid that from happening, we implemented a transfer learning procedure. The network is first trained on the large traditional face recognition dataset LFW, and then finetuned on the training data of the collected dataset in this paper.

4.2 Experiment Result

The experimental results would be posted and discussed in depth in this subsection. We first report the comparison of different algorithm on the collected dataset. And then, we discuss the effect of the chosen of different hyper parameters on the performance.

4.2.1 Compare Experiment Results

Table 1 Table Captions Should Be Placed Above the Tables.

Algorithms	Accuracy	
LeNet [17]	58.0%	
AlexNet [23]	70.7%	
VGG 16 [15]	84.2%	
ResNet 31 [16]	88.5%	
Ours (Res+Attention)	91.3%	

The experimental result is shown in Table 1. It can be seen that our method is able to achieve the best performance 91.3%. When the attention unit is removed, the performance of ResNet backbone dropped to 88.5%. That is because attention unit is able to automatically block out the useless information and focus on the salient local regions of the face image. Without attention unit, the convolutional neural network is much more difficult to extract the useful and salient information and filter out the complex and noisy background.

LeNet [10] only achieves 58.0%. The reason is that LeNet only has two convolutional layers, and therefore it suffers from bad feature extraction ability. The accuracy of AlexNet is 70.7%. That is because AlexNet is deeper than LeNet, and thus has better feature abstraction ability. Accuracy of VGG boosts to 84.2% because of its deeper network and better architecture design. The accuracy of ResNet is able to reach 88.5%, thanks to its residue module.

4.2.2 Parameter Experiment

We mainly discuss the effect of different batch size and learning rate on the performance of the model.

Batch size is important to the training of our model. Therefore, parameter experiments are conducted to evaluated effect of different batch size on the performance, which is shown in Table 1. When batch size is 48, the accuracy is 88.3%. When batch size is increased 64, the accuracy is 90.2%. When batch size is 96, the accuracy is 91.0%. When the batch size is 128, the accuracy is 91.9%. Finally, we double the batch size to 256, and the accuracy is 91.1%. Therefore, we choose 200 as the

batch size for experiment considering the performance and memory usage.

Table 2 Performance of the Model under Different Batch Sizes

Batch size	Training Accuracy
48	88.3%
64	90.2%
96	91.0%
128	91.9%
256	91.1%

Table 3 Performance of the Model under Different Learning Rates

Learning rate	Training accuracy	Test accuracy
0.01	37.3%	29.3%
0.001	91.9%	91.3%
0.0001	77.4%	70/5%

Learning rate also plays an important role in the training of our model. Therefore, we conducted experiments to choose the best learning rate to identify which learning rate is most appropriate. The result is listed in table 2. In this experiment, the batch size is set to 128, and the model is trained for 50 epoch. When the learning rate is 0.01, the training accuracy is only 37.3% and the test recognition rate is also very bad. When the learning rate is 0.001, the training accuracy is above 91% and test recognition rate is 91.3%. When the learning rate is 0.0001, the training recognition rate is above 98% and test sample recognition rate is 23.08% because the algorithm learns too slow.

5. Conclusion

In this work, a new facial recognition method is presented. Our algorithm addresses many issues about facial recognition. Firstly, we collect a large number of samples from internet with different variables such as races, ages and so on. Secondly, we classify each image with different labels; for races, there are White, Black, Asian etc.; or ages, there are 20+, 30+,50+,70+ etc.; for other variables, there are different light conditions, angles of the face and size of the image. Our algorithm addresses these issues very well. We imply some methods in our algorithm and works very well. We also imply attention unit to our algorithm to extract key information and it works very well.

References

[1] Chen Y, Lu X., Wang S. Deep Cross-Modal Image-Voice Retrieval in Remote Sensing (2020). . IEEE Transactions on Geoscience and Remote Sensing,

[2] Burton A M, Wilson S, Cowan M, et al (1999). Face recognition in poor-quality video: Evidence from security surveillance. Psychological Science, vol.10, no.3, pp.243-248.

[3] Timo Ahonen, Student Member, IEEE, Abdenour Hadid, and Matti Pietikainen. Face Description with Local Binary Patterns: Application to Face Recognition, Senior Member, IEEE

[4] John Wright, Student Member, IEEE, Allen Y. Yang, Member, IEEE, Arvind Ganesh, Student Member, IEEE, S. Shankar Sastry, Fellow, IEEE, and Yi Ma, Senior Member, Robust Face Recognition via Sparse Representation , IEEE

[5] Albiol A, Monzo D, Martin A, et al (2008). Face recognition using HOG–EBGM. Pattern Recognition Letters, vol.29, no.10, pp.1537-1543.

[6] Luo J, Ma Y, Takikawa E, et al (2007). Person-specific SIFT features for face recognition, 2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07. IEEE, vol.2, pp. II-593-II-596.

[7] Taigman Y, Yang M, Ranzato M A, et al (2014). Deepface: Closing the gap to human-level performance in face verification, Proceedings of the IEEE conference on computer vision and

pattern recognition. Pp.1701-1708.

[8] Gao S, Tsang I W H (2010), Chia L T. Kernel sparse representation for image classification and face recognition, European conference on computer vision. Springer, Berlin, Heidelberg, pp.1-14.

[9] Wagner A, Wright J, Ganesh A, et al (2011). Toward a practical face recognition system: Robust alignment and illumination by sparse representation. IEEE transactions on pattern analysis and machine intelligence, vol.34, no.2, pp. 372-386.

[10] Yang M, Zhang L (2010). Gabor feature based sparse representation for face recognition with gabor occlusion dictionary, European conference on computer vision. Springer, Berlin, Heidelberg, pp.448-461.

[11] Sun Y, Wang X, Tang X (2014). Deep learning face representation from predicting 10,000 classes, Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1891-1898.

[12] Ouyang W, Wang X, Zeng X, et al (2015). Deepid-net: Deformable deep convolutional neural networks for object detection, Proceedings of the IEEE conference on computer vision and pattern Recognition. pp.2403-2412.

[13] Deng J, Guo J, Xue N, et al (2019). Arcface: Additive angular margin loss for deep face recognition, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4690-4699.

[14] Wang H, Wang Y, Zhou Z, et al (2018). Cosface: Large margin cosine loss for deep face recognition, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5265-5274.

[15] Simonyan K, Zisserman A (2015). Very deep convolutional networks for large-scale image recognition. International Conference on Learning Representations, pp.298-302.

[16] He K, Zhang X, Ren S, et al (2016). Deep residual learning for image recognition, Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770-778.

[17] LeCun Y (2015). LeNet-5, convolutional neural networks. URL: http://yann. lecun. com/exdb/lenet, vol.20, no.5, pp.14.

[18] Parkhi O M, Vedaldi A, Zisserman A. Deep face recognition. 2015.

[19] Huang G B, Mattar M, Berg T, et al (2008). Labeled faces in the wild: A database forstudying face recognition in unconstrained environments.

[20] Kemelmacher-Shlizerman I, Seitz S M, Miller D, et al (2016). The megaface benchmark: 1 million faces for recognition at scale, Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4873-4882.

[21] Liu Z, Luo P, Wang X, et al (2015). Deep learning face attributes in the wild, Proceedings of the IEEE international conference on computer vision. pp. 3730-3738.

[22] Zhang Z, Luo P, Loy C C, et al (2014). Facial landmark detection by deep multi-task learning, European conference on computer vision. Springer, Cham, pp.94-108.

[23] Krizhevsky A, Sutskever I, Hinton G E (2012). Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems. pp. 1097-1105.