Facial Expression Recognition based on Mini_Xception

Sizhen Lyu *

College of Information Science and Technology, Jinan University, Guangzhou, 510632, China

* Corresponding author email: cap14@stu2020.jnu.edu.cn

Abstract. As the premise of emotion recognition, facial expression recognition has received extensive attention and discussion, which aims to capture face information by computer, understand and classify it according to the way of thinking of people to improve the effect of human interaction. Through previous efforts have significantly improved the accuracy, few of them consider the application effect in specific scenarios. In this paper, we make a summarize of the existing research methods, and explore the generalization ability of the network model in different scenarios based on the MiniXception network. We also explore the scope with smaller or bigger impact on network performance, and make assumptions and prospects for the optimization method and development direction in this scenario.

Keywords: Facial Expression Recognition; Mini_Xception; Deep Learning.

1. Introduction

With the continuous development of computer science, our lives have also been greatly affected, the standards of demands to more efficient way of human-computer interaction models have become higher and higher. Emotion recognition as a basis of human-computer interaction has gradually become a research hotspot, and as a premise of emotion recognition, facial expression recognition, has also received a widespread attention and discussion. Facial expression recognition aims to capture facial information by computers and understand and classify it according to the way as human thinking, which makes a better human-computer interaction effects and strengthens the intelligence of computers. Facial expression recognition can be widely used in robot manufacturing, medical, safety, education and many other fields which can all receive a great convenience from the support of this technology.

The early facial expression recognition methods are generally based on machine learning, whose main process is to preprocess the input picture to manually extract features and finally classify the image according to the information gained by feature extraction. However, the work of selecting features in traditional classification methods is complex, and the selection of these hand-crafted feature has a great impact on the classification effect. Although it can obtain a certain degree of recognition effect, the accuracy of traditional classification methods is far behind the application requirements of expression recognition. With the establishment of facial expression database and the inventing of neural network framework with increasingly ability of computing, the latter, deep learning [1] has gradually become the mainstream in facial expression recognition. Breuer et al. used CNN visualization technology to extract features [2]. Zhang et al. proved that the insurance of model's convergence would be realized with task-wise early stopping [3]. Thanks to the robust ability of representing feature of CNN, the methods developed from deep learning have significantly improved the accuracy compared with the traditional machine learning manual feature extraction method. However, we argue few of them pay attention to the application effect in specific scenes. This paper, aiming for exploring the generalization ability of expression recognition systems in different scenes, adopts a lightweight and simple network structure for feature extraction and model training in order to reduce the cost when using the model, and then designs experiments to analyze the research subjects.

In the specific work we carry out, we build a facial expression recognition model with Mini_Xception as the foundation and validate the feature extraction ability of our structure. We find the optimal dimension setting and parameter combination by adjusting network dimension and

training parameters, which is regarded as the basis for subsequent experiments. Then we select some images from dataset randomly and adjust their angle and brightness to different degrees before sending them into model for classifying. As obtaining the result of recognition, we can analyze the model's performance in different application scenarios, by which we figure out that the network model we use has a good generalization ability in dark conditions, while it's weak in brighter conditions, additionally, its adaptability to angle is also just acceptable in minor changes.

The assignment of paper is not that complicated. In Section 2, methods which have already been proposed will be introduced in detail, including a brief introduction of deep learning, the reviewing of CNN, the development of Inception and the specific use of Mini_Xception expression recognition. Section 3 is about research experiment, which tests the performance of our model in different network dimensions and parameter combinations to obtain the best configuration. The expression recognition effect of the model under different brightness and angles are also be explored. Section 4 is left for discussion, in which we make a summarize about some problems in the topic that needed to be solved, and separately propose ideas and expectations from the perspective of application and technology.

2. Methods

2.1 Revisiting Convolutional Neural Network

Deep learning belongs to the category of machine learning, comparing to shallow machine learning which classification is based on manually extracted features, deep learning mainly realizes the automation of feature extraction and classification of traditional machine learning, which greatly reduces the time and labor cost in the feature extraction stage, and also lower the threshold of machine learning. In deep learning, the hidden layer of network performs a linear combination of the data input, and the weight between the input layer and the hidden layer is equivalent to the coefficient of the variable in the linear combination. After countless attempts, the neural network can adjust the parameters to the combination most consistent with the current input and expected output, which is the process of automatic feature extraction in deep learning.

Convolutional neural network [4] (CNN) has a significant effect on the extraction of image semantic information, which mainly consists of three parts: convolutional layer, pooling layer and fully connected layer. First, the convolutional layer performs the main convolution operation, which can obtain the feature correlation on the adjacent positions of one single picture, and it can share a set of weight combinations in the same feature map, which greatly reducing the amount of parameters to be considered. After the convolutional layer here comes to the pooling layer, which is mainly responsible for extracting the features obtained from the convolutional layer as expected, it integrating several feature maps obtained after the former operation, reducing the size and computational complexity of the feature map, and effectively avoiding the premature fitting of neural networks. Common nonlinear sampling operations include maximum pooling and average pooling. At last, the fully connected layer (also known as the activation layer) is used to classify the features which were extracted previously, reduce the dimension of the multidimensional feature map to a one-dimensional vector, and then map to each expected situation to achieve the effect of a classifier.

2.2 Development of Inception

As the problems is getting more and more complex, the number of layers in a convolutional neural network continues to increase, the problems of vanishing gradient and model degradation appear correspondingly. Since then, batch normalization is used to place the data, which is going to be the input of the activation function, in the gradient unsaturated area to alleviate the problem of gradient disappearance. However, the degradation of network performance brought by batch normalization is not supposed to be ignored. The residual network adds the input and output through a bypass directly, which further alleviates the problem of vanishing gradient and model degradation. It makes the network structure of neural networks becomes deeper and deeper while still maintaining a good performance, such as ResNet [5] and DenseNet [6].

Knowing that the most efficient way to improve the performance of a neural network is to increase the depth or the width of network. In a certain degree, ResNet gives an answer to the problem of network vertical development. Then Inception is proposed to increase the network width while avoiding the appearance of a large number of parameters. It combines multiple convolution and pooling operations together to form a network module, then spliced them into an overall network structure, which is designed to be sparse while being able to produce dense data, so it can not only increase the neural network performance, but also ensure the efficiency of computing resources. Google Labs put forward GoogleNet [7] (Inception v1) in 2014 and continuously developed it in the next two years.

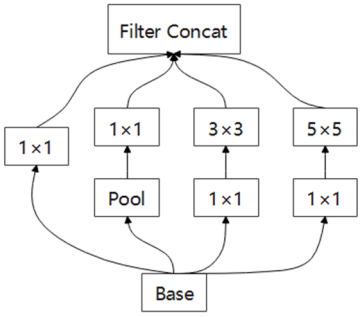

Fig 1. The structure of Inception v1

Since then, Google Labs has successively designed Inception v2 [8] and Inception v3 [9] in 2015. Inception v2 has made an introduction of batch normalization, which has brought profound significance to the development of deep learning. In the meanwhile, it has substituted the 5×5 convolution in v1 with two connected 3×3 convolution cores. This connection method reduces the number of parameters while maintaining the range of receptive fields, and can avoid expression bottlenecks, strengthen the ability of nonlinear expression. Inspired by the idea of shortcut in ResNet, asymmetric convolution is considered in Inception v3 to replace the convolution of $n \times n$ with a module consists of $1 \times n$ followed by $n \times 1$. However, this decomposition effect is not ideal in the early use of the network, and it is more suitable for medium size feature maps.

In the same year, Google proposed Xception [10] based on Inception v3, which has less computational complexity and is easy to migrate while maintaining a high accuracy. As the Figure 2 shown, Xception mainly uses the depthwise separable convolution [11], which breaks down the traditional operation of convolution, dividing the complete step into two stages as followings. Assuming the size of initial convolution core is 3×3, the operation of depth separable convolution can be described as following: Supposing there are N input feature maps, we first use N convolution kernels to convolve each single feature map, by which we collecting N results, while features in different channels but the same spatial position haven't been extracted, so after that, N convolution layers with 1×1 convolution kernel are used to convolve the feature maps output by the last step, finally N results are obtained, which improves the effect of the network model without increasing network complexity.

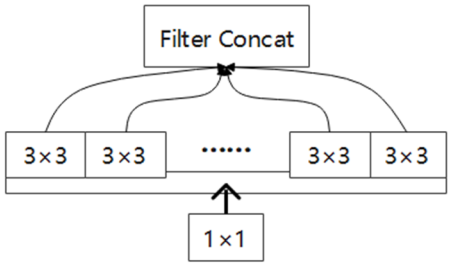

Fig 2. The basic structure of Xception

2.3 Mini_Xception

Inspired by the architecture of Xception, Mini_Xception [12] uses light CNN to ensure performance, considers not only residual modules but also depthwise separable convolutions. On the basis of removing the final full connection layer, the number of parameters is further reduced by eliminating the parameters in the convolution layer. Follow the idea of ResNet, residual connections are added to the network structure to improve the accuracy. The whole process includes four depthwise separable convolution with residual branch, batch normalization (BN) and ReLU are used after each convolution to ensure the gradient in forward propagation. Finally, the global average pooling is used to calculate the average of overall feature map to get the result, which is immediately send into softmax for classification. The basic structure of Mini_Xception is shown in Figure 3.

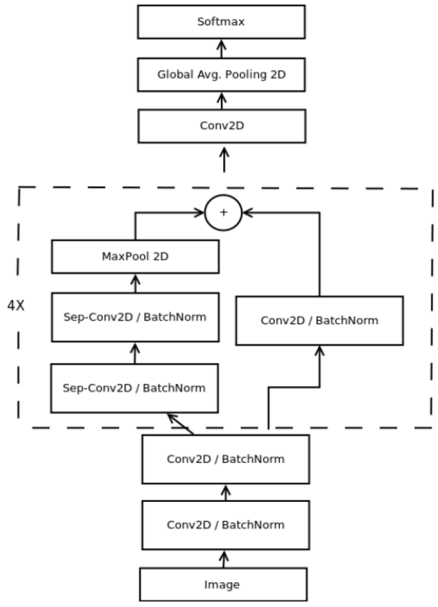

Fig 3. The basic structure of Mini_Xception[13]

2.4 Expression Recognition based on Mini_Xception

As the Figure 4 shown, the process of expression recognition by Mini_Xception is as following. First, input a single image into two ordinary convolution pooling modules which is composed of 3×3 convolution kernels, BN and ReLU for spreading and preliminary feature extraction. Then the obtained image features (supposing there are X feature maps obtained) are feed into the residual extraction module. The main branch mainly uses depthwise separable convolution to extract features: first, set the number of groups exactly the same as the number of input channels (X), then use 3×3 convolution kernels to make one-on-one depthwise convolution to obtain the global information of expression features, and then use 1×1 convolution to realize pointwise convolution on the feature maps obtained in depthwise convolution, thus, different spatial information on the same spatial location has already been integrated into the new feature map. The number of output feature maps is consistent with the number of pointwise convolution kernels. As things happening in the main branch, the residual branch raises the dimensions of X feature maps by 1×1 convolution kernel and processes batch normalization. The result of main branch and residual branch will be added together as the next module's input when the dimensions in residual branch reach the same level as the main branch. After extracting feature of four times travelling in residual extraction module, the high-dimensional feature map gained before will be mapped to 7 expression tags through the 3×3 convolution kernel dimension reduction processing. After an adaptive average pooling operation, a tensor is obtained to express the expected degree of each expression. Finally, the classification results are obtained through softmax.

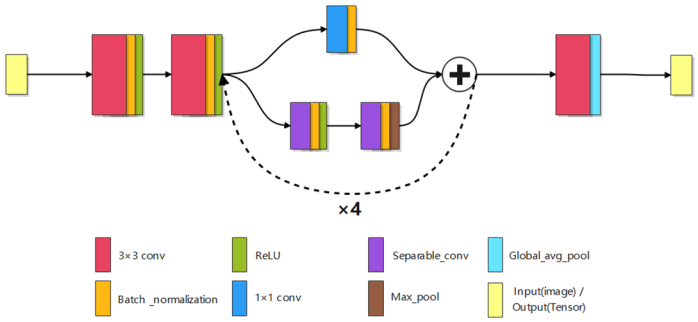

Fig 4. The framework of expression recognition based on Mini_Xception

3. Experiments

3.1 Dataset

The database selected for facial expression recognition has a significant impact on the effect of recognition. It's important to try to ensure that the selected dataset not only has enough training data, but also contains as many different scenes as possible. FER2013 [14] is selected for following experiments, which is an extensive unconstrained database automatically collected by Google Image Search API introduced in the ICML representative learning challenge in 2013, thus becoming one of the benchmarks for comparing the performance of expression recognition models and being selected as the data source of the Kaggle face recognition competition in 2013. Error mark frames are excluded automatically, and each picture is adjusted and clipped to a gray image of 48×48 pixels. According to the 8:1:1 distribution standard, Fer2013 includes 28709 images for training, 3589 images in publictest and privatetest. Each image has an expression tag. There are totally seven emotion tags in the entire dataset, from 0 to 6 representing "Angry, Disgust, Fear, Happy, Sad, Surprise, Neutral".

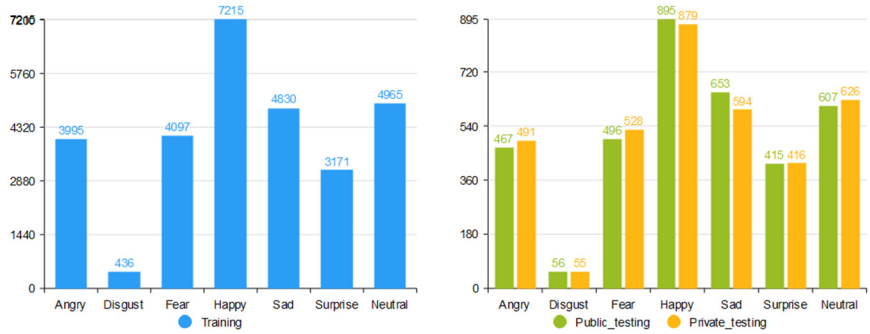

Fig 5. The distribution form of Fer2013

The distribution of Fer2013 is shown in Figure 5. Fer2013 has collection errors, and the accuracy of human is only 65%+- 5%. In addition, the label disgust contains too few samples comparing with others, so the model has congenital defects and deficiencies in the recognition of disgust.

3.2 Evaluation Indicator

In the evaluation of the model performance, accuracy, precision and recall [15] are always used to describe if the performance of a model is good or not. First, we will review the terms used to describe these three attributes. We define the classes we care about as "positive classes" and the rest as "negative classes". With the using of classifier, a sample can be described by four types of identities: real positive class, real negative class, classifier positive class and classifier negative class. Then, according to whether the classifier's prediction on the test dataset is correct or not, all samples can be divided into the following four sets: TP (judge the real positive class successfully), FN (judge the real positive class as an opposite one), FP (judge the real negative class as an opposite one) and TN (judge

the real negative class successfully). T and F represent whether the classifier result is correct; P and N, representing positive and negative classes.

Accuracy indicates the proportion of the correct sample among all samples, which can be calculated as:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Precision represents the proportion of real positive classes are there in the samples classified as positive classes by the classifier, which means how many correct positive classes the classifier has judged. Recall indicates the proportion of all real positive classes that have been successfully recalled by the classifier, in a word, it shows the amount of real positive classes have been found by the classifier. The precision and recall can be calculated through formula (2) and (3), respectively.

$$\text{precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (2)$$

$$recall = \frac{TP}{TP+FN} \quad (3)$$

Obviously, precision and recall rate are standard for binary classification. While in the context of our expression recognition model, which is a multi-classification model, since different expressions are equivalent to each other, macro coverage is used for calculating the value of precision and recall for each expression tag, and then take their average as the evaluating indicator.

3.3 Experiment Design and Setteing

Here are the purposes of the experiments:

Experiment 1: With the structure of network staying still, change the dimension of network to explore how our network preforms in different dimensions.

Experiment 2: On the premise of the dimension obtained through experiment 1, we adjust the hyperparameters and figure out the changes of network performance under different combinations of hyperparameters to find the best hyperparameters combinations for the following Experiment 3.

Experiment 3: As getting a model with good performance under the exploration of Experiment 1 and 2, we use this model to test its generalization ability of facial expression recognition in different scenes such as brightness and rotation angle.

To realize these purposes, we implement Mini_ Xception on pytorch, and then train the model on Fer2013's training dataset, make evaluation to the performance on Fer2013's privatetest set, find an appropriate network dimension and parameter combination. After that, we find some images that can be correctly identified by the model on Fer2013's publictest set, use matlab to do the adjustment of brightness angle rotation, and then use the model to identify them again, record the accuracy and analyze the performance. During the experiments, all the operation of training, testing and recognition were carried out on a 3070GPU.

3.4 Performance Analysis

3.4.1 Effect of Different Dimensions

We first explore how network preforms in different dimensions when hyperparameters are fixed, whose results can be seen in Table 1. Here, we set a batch size of 40, learning rate of 0.01, and epoch of 100.

Table 1. Network performance with different dimensions

Dimension	Accuracy	Precision	Recall	Time/epoch
8~128	50.6%	44.3%	51.2%	58.3s
64~1024	53.6%	60.6%	52.2%	146.3s

In the residual extraction module, the network performs better after increasing network dimension. The accuracy and precision rate have been significantly improved. Although the time consumption has also increased, the loss is acceptable with the significant improvement of performance. The reason why is that the higher dimension group is more detailed in the initial feature extraction of the image

than the former one. Before the image enters the residual extraction module, the higher dimension group will successively use 96 convolution kernels to expand the single input grayscale image to 64 output feature images. Compared with the 8 outputs of the lower dimension group, the former one will undoubtedly describe comprehensively the input image, namely, the facial expression information, in more detail; The higher dimension in the residual extraction module also represents more convolution kernels, which means more sufficient and detailed feature learning, and the increase in time consumption is still acceptable, so the latter group is more worthy of being used as the network structure for subsequent experiments.

3.4.2 Hyperparameter Comparison Experiment

When the initial batchsize is 40, the decrease of train loss is not fast enough, and the convergence effect is not satisfactory. Since dataset is large enough, then batchsize can be appropriately reduced to 15, if it continues reducing, which will lead to the non-convergence of the model; when learning_ rate is 0.01, it will be found that the final performance we obtain is not as accurate as the temporary accuracy rate if we observe the information output after each epoch. It is speculated that the model may jumped out of the point where the model performs best, and the learning rate is too slow to wait for the 'Reduce Learning Rate on Plateau' to gradually adjust, so the initial learning rate is set to 0.001. Although a certain amount of time has been sacrificed, the problem has been solved better; The value of epoch was established on the basis of many failed experiments end up with over fitting. During the experiment, it was found that the model was basically over fitted after 200 epochs. Therefore, 100 epochs were selected to be the initial value of epoch for the experiment. When the batchsize was small enough and there was little room for learning rate to decline, epochs were added appropriately to make the training more adequate, and finally the accuracy rate has achieved 61.9%. Therefore, the fourth combination in the table 3 was selected as the hyperparameters configuration of the subsequent experiment.

Table 2. Network performance in different hyperparameters

Combination	Batchsize	Learning_rate	Epoch	Accuracy
C1	40	0.01	100	53.6%
C2	15	0.01	100	56.7%
C3	15	0.001	100	60.1%
C4	15	0.001	150	61.9%

3.4.3 Exploration of Model Generalization Ability

We selected some face images in publictest set of Fer2013 which have already successfully classified by our model. To ensure fairness and persuasion, we selected three images from every emotion tag which leads to a total number of 21, and then used matlab to adjust their face angles. We also use matlab to enhance or weaken the brightness of the face image. The images with various angles and brightness are visualize in Figure 6.

(a) Adjusting angles of face

(b) Adjusting brightness of face

Fig 6. Images with various angles and brightness

According to following Table 3, it can be found that the model still has some adaptability to the face rotated by 1 °~15 °; Considering the accuracy rate of 60% of the model itself, the final accuracy

rate of the model is meaningless after the rotation of 20 °~30 °, when the range of rotation comes to 45° or even higher, the model basically loses its ability of judgment. Therefore, the generalization ability of this model is just general.

Table 3. Performance in different angles

Rotation(degree)	Accuracy
1	95.2%
4	90.5%
8	80.9%
10	76.2%
15	57.1%
30	42.9%
45	28.6%

From the Table 4 below, it can be seen that the model has a certain discrimination ability within the range of 0.3~1.5 times of the original image brightness. Considering that the judgment ability of the model itself is about 60%, the accuracy rate of 61.9% is already low enough when the brightness is 1.4 times of the original image, and result given by the model is meaningless when the brightness is more than 1.5 times or higher of the original image. Overall speaking, the model discrimination ability is stronger when the brightness is comparatively dark than when the brightness is comparatively bright. The reason is that in the process of brightness enhancement, due to the limitation of 0~255 pixel values, it is easy to form a bright spot consists of all 255 locally, and the relationship between pixel points is missed, which greatly hinders the recognition of facial structure and expression; While in the process of dimming, the pixel value will not reduce to 0 because it is adjusted by multiple, and the relationship between pixels is not easy to be destroyed, so the performance is more stable than the performance in lighting up faces.

Table 4. Performance in different brightness

Brightness(times)	Accuracy
0.3	66.7%
0.5	76.2%
0.7	85.7%
0.9	95.2%
1.1	100.0%
1.2	85.7%
1.3	80.9%
1.4	61.9%
1.5	57.1%
1.8	14.3%
>2.0	<9.5%

4. Discussion

Since the complexity of human expression language, facial expression recognition is inherently difficult to achieve a high accuracy. In order to further improve the emotional intelligence of machine, we need to carry out more detailed research and more thorough division of expressions. For example, in addition to the seven expression tags mentioned in this paper, we need to train our models to recognize composite expressions such as pleasantly surprised, and micro expressions such as mouth tilt upwards slightly at the corners as well. If conditions allow, we need to combine voice and body language to jointly judge to achieve more accurate emotional recognition.

Although the Fer2013 dataset has sufficient training data, but the number of expression ‘disgust’ is far from enough, so using GAN to generate a small number of samples to balance should be

considered. In addition, the single scene also leads to a poor effect of the model in practical application. For example, the change of lighting scene in natural conditions is even much more complex than we adjusted in this experiment, local shadows or highlights will bring more difficulties to expression recognition, which requires more detailed scene factors to be considered in the range of training.

In addition, there are also some problems to be solved in practical applications, for example, people in different regions have different facial structure and expression habits, which may need more systematic collection of expression information of people from different regions in the future to achieve more accurate and humane recognition results; Another example is the poor robustness of the model, although Mini_ Xception has reduced the size of the model as much as possible, but the accuracy is still not high enough, and its generalization ability is also insufficient. More lightweight and efficient model development is still one of the directions of future efforts. Besides, the reduction of hyperparameters, the enhancement of robustness and interpretability are all conducive to the implementation and application of the method.

5. Conclusion

In this paper, we explained the research background and significance of facial expression recognition, and did corresponding experiments to analyze the performance of our model in different scenarios such as angles and brightness. The process of carrying out the work is as following: First we compare the characteristics of traditional methods and methods in deep learning which is more advance, and then we focus on the field of deep learning; After reviewing the convolutional neural network, we make a comparison of the existing networks with good performance, then the relatively lightweight and simple MiniXception was selected as the network model used in our experiments. In the experimental stage, after introducing Fer2013 dataset, the design and content of concrete experiments were described, we ensure the network structure and parameter configuration of the network, which is used to identify and classify facial expressions at different angles and brightness. After that, the experimental results were analyzed which leads to a conclusion that our network model has good generalization ability for rotation angle and brightness in a certain range of adjustment, and there is an urgent need for certain methods to maintain the performance of the model when these changes exceed their range.

References

[1] LECUN Y, BENGIO Y, HINTON G. Deep learning[J]. nature, 2015, 521(7553):436-444.

[2] Breuer R, Kimmel R. A deep learning perspective on the origin of facial expressions[J]. arXiv preprint arXiv:1705.01842, 2017.

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

[4] LECUN Y, et al. Lenet-5, convolutional neural networks[J].

[5] Targ S, Almeida D, Lyman K. Resnet in resnet: Generalizing residual architectures[J]. arXiv preprint arXiv:1603.08029, 2016.

[6] Iandola F, Moskewicz M, Karayev S, et al. Densenet: Implementing efficient convnet descriptor pyramids[J]. arXiv preprint arXiv:1404.1869, 2014.

[7] Anand, R., et al. "Face recognition and classification using GoogleNET architecture." Soft computing for problem solving. Springer, Singapore, 2020. 261-269.

[8] Halawa L J, Wibowo A, Ernawan F. Face recognition using faster R-CNN with inception-V2 architecture for CCTV camera[C]//2019 3rd International Conference on Informatics and Computational Sciences (ICICoS). IEEE, 2019: 1-6.

[9] Tio A E. Face shape classification using inception v3[J]. arXiv preprint arXiv:1911.07916, 2019.

[10] Chollet, François. "Xception: Deep learning with depthwise separable convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[11] CHOLLET F. Xception: Deep learning with depthwise separable convolutions[C]//CVPR. 2017:1800-1807.

[12] Fatima S A, Kumar A, Raoof S S. Real time emotion detection of humans using mini-Xception algorithm[C]//IOP Conference Series: Materials Science and Engineering. IOP Publishing, 2021, 1042(1): 012027.

[13] Arriaga O, Valdenegro-Toro M, Plöger P. Real-time convolutional neural networks for emotion and gender classification[J]. arXiv preprint arXiv:1710.07557, 2017.

[14] Zahara L, Musa P, Wibowo E P, et al. The facial emotion recognition (FER-2013) dataset for prediction system of micro-expressions face using the convolutional neural network (CNN) algorithm based Raspberry Pi[C]//2020 Fifth international conference on informatics and computing (ICIC). IEEE, 2020: 1-9.

[15] Juba B, Le H S. Precision-recall versus accuracy and the role of large data sets[C]//Proceedings of the AAAI conference on artificial intelligence. 2019, 33(01): 4039-4048.