Hindawi
Security and Communication Networks
Volume 2021, Article ID 9539022, 10 pages
https://doi.org/10.1155/2021/9539022

Research Article

Real-Time Facial Expression Recognition System for Video Big Sensor Data Security Application

Zhi Yao[1], **Hailing Sun**[1] and **Guofu Zhou**[1,2]

[1]*Guangdong Provincial Key Laboratory of Optical Information Materials and Technology & Institute of Electronic Paper Displays, South China Academy of Advanced Optoelectronics, South China Normal University, Guangzhou 510006, China*
[2]*Shenzhen Guohua Optoelectronics Technology Co., Ltd., Shenzhen 518110, China*

Correspondence should be addressed to Hailing Sun; sunsmile1225@163.com

Received 24 May 2021; Accepted 8 August 2021; Published 20 August 2021

Academic Editor: Youwen Zhu

Facial video big sensor data (BSD) is the core data of wireless sensor network industry application and technology research. It plays an important role in many industries, such as urban safety management, unmanned driving, senseless attendance, and venue management. The construction of video big sensor data security application and intelligent algorithm model has become a hot and difficult topic in related fields based on facial expression recognition. This paper focused on the experimental analysis of Cohn–Kanade dataset plus (CK+) dataset with frontal pose and great clarity. Firstly, face alignment and the selection of peak image were utilized to preprocess the expression sequence. Then, the output vector from convolution network 1 and β-VAE were connected proportionally and input to support vector machine (SVM) classifier to complete facial expression recognition. The testing accuracy of the proposed model in CK + dataset can reach 99.615%. The number of expression sequences involved in training was 2417, and the number of expression sequences in testing was 519.

1. Introduction

Video is the best data type with largest amount and high degree of industrialization in BSD applications. Facial data is becoming more important for research. Facial expression recognition and analysis is the basic supporting technology of the above applications. It has become a hot and difficult topic to construct a security application and intelligent algorithm model of video BSD based on facial expression recognition [1, 2]. The main task of facial expression recognition is to realize automatic, reliable, and efficient facial information extraction and recognition.

The research direction of facial expression recognition includes hardware detection and programming recognition. Turabzadeh et al. achieved accuracy of 47.44% in automatic emotional detection by Field Programmable Gate Array (FPGA) [3]. Then, Mehta et al. realized emotion recognition in augmented reality (AR) by Microsoft HoloLens (MHL) [4]. The programming direction was aimed at optimizing the structure of model. There were lots of datasets in the field of facial expression recognition, such as dataset Multi-PIE and CK+ [5, 6]. Mao et al. utilized the Kinect sensor to track facial action unit (AU) and feature point position (FPP) to achieve about 80% accuracy [7]. In addition, Yang et al. achieved accuracy of 97.3% by extracting the information of expression components on CK+ dataset [8]. Zhang et al. proposed an end-to-end deep learning model that utilized GAN network to achieve 91.8% accuracy on Multi-PIE dataset [9].

Some scholars had also carried out experiments from other research directions except for the optimization of network [10]. Meng et al. achieved recognition accuracy of 95.37% on CK+ dataset as well as maintaining identity characteristics [11]. Cheng and Wang considered that facial expression recognition should be analyzed differently in different fields [12]. Facial expression recognition and psychology were very popular in recent years. Picard proposed the idea of emotional computing in 1997 [13]. Ekman

and Friesen proposed Facial Action Coding System (FACS) to define facial emotions [14]. Geometric analysis and appearance analysis were applied to facial expression recognition on this basis [15].

Nowadays, the research of facial expression recognition involved deep learning, psychological analysis, physiological analysis, education, and other disciplines [16]. Its application and research had gradually expanded to three-dimensional and multiangle directions [17]. However, the low efficiency of the above methods and excessive facial influencing factors remained to be resolved [18]. In this paper, face alignment and the selection of peak image were utilized to enhance data features. Then, the output vector from convolution network 1 and β-VAE were connected proportionally and input to SVM classifier to complete facial expression recognition. The proposed model has advantages in accuracy compared with related models.

2. Materials and Methods

The recognition of facial expression sequence described in this paper is mainly divided into three processes: preprocessing of expression sequence, feature extraction of expression sequence, recognition, and classification of expression sequence.

2.1. Preprocessing of Expression Sequence. The CK+ dataset including frontal face is utilized in this paper. The dataset includes 123 themes, 593 sequences, and 7 expression labels. Expression labels consist of anger, contempt, disgust, fear, happiness, sadness, and surprise. It is a pity that only 327 of the 593 expression sequences have expression labels, which are the research object of this paper. The significant reason for choosing CK+ dataset is that the expression label corresponds to an expression sequence rather than a single expression. In addition, the main application scene of the proposed model is the standard frontal expression recognition.

The dataset has the disadvantages of small amount and irrelevant noise although it has the above advantages and is preprocessed to improve the accuracy of facial expression recognition and the stability of training process. This section mainly consists of face alignment of images and selection of expression peak images.

2.1.1. Face Alignment of Images. The original dataset has standard facial pose, but it is inevitable to have redundant image noise. Therefore, multitask convolutional neural network (MTCNN) is utilized to align the image [19].

The principle of MTCNN is that the matching of multiple regression frames can accurately locate the frame of input face image. MTCNN is designed by the cascade of proposal network (P-Net), refine network (R-Net), and output network (O-Net) in terms of network structure. Firstly, the input image is input to P-net, and the output vector shown in Figure 1 is obtained subsequently. Its length is 2, 4, and 10, respectively, representing the classification score of bounding boxes (bbox), the offset of bbox, and the value of landmark. In P-net, the classification score is utilized to select initial bbox, whose specific location is calibrated by the offset of bbox. Then, these bboxes are selected based on Intersection over Union (IoU). The threshold of IoU is set in advance. The repetition of this operation can eliminate massive overlapping bbox. Finally, the reserved bbox is moved based on coordinates and modified to enter R-Net.

The output type of R-net and O-net is the same as P-Net's as shown in Figure 1, and their goal is to further adjust the size and location of the bounding box. MTCNN can get the accurate bbox and landmark coordinates of input image on the premise of avoiding deformation and retaining more details.

The role of MTCNN in this paper is aimed at completing accurate face alignment, so the landmark is not utilized. The diagram of IoU is shown in Figure 2.

The two rectangular frames shown in Figure 2 are predicted box and ground truth, respectively. In each output of network, the coordinate of bbox corresponding to the highest value of score is intersected and calculated to find the IoU. The value of IoU is compared and selected with the value of threshold, and the maximum score is saved and transferred [20].

2.1.2. Selection of Expression Peak Images. The basis of the operation in this section is that the expression sequence contains initial calm expression and peak expression. But only the peak expressions are processed when the expression sequence feature is extracted. Therefore, the structural similarity (SSIM) is utilized to calculate similarity to complete the selection of peak images [21].

SSIM is an index utilized to measure the similarity of images and is utilized to judge the structural similarity between two images. The SSIM of two images includes brightness comparison, contrast comparison, and structure comparison [22]. Firstly, the average gray value of image is calculated, as shown in the following equation:

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i. \tag{1}$$

x_i represents the pixel value of the image and N represents the size of window utilized in calculation. A window of $[N \times N]$ is taken from the image every time, and it will be continuously slid in the calculation process of SSIM. The overall SSIM of the image will be obtained by averaging the local value of SSIM. Then, the average gray μ_x can be utilized to calculate the gray standard deviation σ_x in the following equation:

$$\sigma_x = \left(\frac{1}{N-1}\sum_{i=1}^{N}\left(x_i - \mu_x\right)^2\right)^{1/2}. \tag{2}$$

After basic parameters are obtained, the three image comparison indices mentioned above are calculated, respectively, as shown in the following equations:

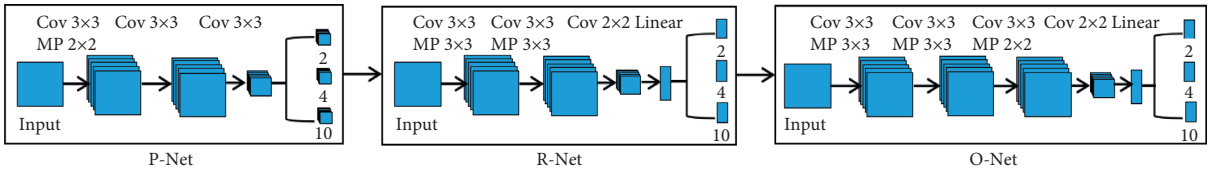

Figure 1: Flowchart of face alignment.

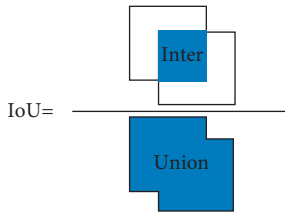

Figure 2: IoU calculation principle diagram.

$$I(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \tag{3}$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \tag{4}$$

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}. \tag{5}$$

C_1, C_2, and C_3 are constants. They are utilized to avoid the denominator of fraction approaching 0. The value of SSIM can be obtained by multiplying the above three indicators, as shown in the following:

$$\mathrm{SSIM}(x, y) = \frac{\left(2\mu_x\mu_y + C_1\right)\cdot\left(2\sigma_{xy} + C_2\right)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\cdot\left(\sigma_x^2 + \sigma_y^2 + C_2\right)}, \tag{6}$$

where C_3 in is represented by C_2; then, the similarity between images can be measured by SSIM. The rules defined in Algorithm 1 are utilized in experiment when peak images are selected.

The dataset utilized by the model in this paper can be selected from each expression image sequence after the parameters p_1 and p_2 are obtained.

2.2. The Principle of Feature Extraction. The ultimate goal of this paper is aimed at classifying the CK+ dataset exactly. Therefore, the task of this section is to fully extract features of input facial expression images. Features extracted by the proposed model consist of two parts. The first part is facial texture features extracted by the network 1 composed of 2D convolutional network. The second part is generative features extracted by β-VAE autoencoder.

2D convolution has universality for feature extraction. An example on CK+ dataset is shown in Figure 3.

2D convolution is developed from one-dimensional signal convolution. 2D image convolution changes the direction of one-dimensional convolution into the width and height directions of image at the same time. Therefore, 2D image convolution is performed on the pixels of image.

The convolution kernel with a specified template size is utilized to perform sliding convolution. The convolution process shown in Figure 3 performs multiplication and addition of parts with the same value. The distribution of values in convolution kernel can represent the distribution of pixels in the window of image. Pixels of expression image shown in Figure 3 are an enlarged representation rather than real situation. The structure of network 1 of the proposed model is shown in Figure 4.

The reason why grayscale image is utilized instead of RGB is that there are differences in the background brightness of the image on CK+ dataset. This effect can be reduced by image graying and face alignment described in Section 2.1.1.

In addition to network 1, the proposed model introduces β-VAE network to extract the generative features of facial images. VAE network is a network modification of codec, whose feature vectors of middle layer can be expressed in the form of Gaussian distribution. This is because the superposition of Gaussian distribution can fit any distribution, which makes the VAE network more representative. The network structure of codec is shown in Figure 5 [23].

$q(\cdot)$ is the distribution function of the encoding part, and $p(\cdot)$ is the distribution function of the decoding part. The process of codec is aimed at training the distribution of coding network and decoding network. The intermediate vector z can well represent the generative features of input image when restored image x' and input x are infinitely close. Each sample z corresponds to a set of μ and σ for VAE. The sum of all Gaussian distributions in integration domain is the original distribution $p(x)$ and objective function equation:

$$p(x) = \int_z p(z)p(x|z)\mathrm{d}z. \tag{7}$$

$p(x)$ can be expressed as follows by introducing logarithmic computation:

$$\begin{aligned}\log p(x) &= \int_z q(z|x)\log p(x)\mathrm{d}z \\ &= \int_z q(z|x)\log\left(\frac{p(z, x)}{q(z|x)}\right)\mathrm{d}z + \int_z q(z|x)\log\left(\frac{q(z|x)}{p(z|x)}\right)\mathrm{d}z.\end{aligned} \tag{8}$$

The second term of (8) can be defined in the following equation based on the definition of divergence:

Input: image similarities after ranking in sequence S, similarity reference value s_0, the number of images in sequence n, images i_1, images i_2, the number t, similarity between i_1 and i_2 s_{12}, and flag parameter f.
Output: the starting point where the peak image appears p_1, The ending point where the peak image appears p_2.

```
(1)  for all images ∈ image sequence do
(2)    t = 0, f = 0
(3)    S = similarities of images sequence
(4)    if S[−1] − S[0] > 0.2 then
(5)      s0 = S[0] + 0.15
(6)    else if S[−1] − S[0] < 0.1 then
(7)      s0 = S[0] + 0.03
(8)    else
(9)      s0 = S[0] + 0.05
(10)   while t < n − 1 do
(11)     i1 = image [t]
(12)     i1 = image [t + 1]
(13)     if s12 < s0 and f = 0 then
(14)       p1 = t + 1
(15)       f = 1
(16)     else if s12 > 0.915
(17)       p2 = t
(18)       f = 1
(19)     else
(20)       p2 = n − 1
(21)     t = t + 1
(22)   if p2 − p1 > round((2/5)n) then
(23)     p1 = round((2/5)n) − 1
```

Algorithm 1: Rule for image sieving.

Input image

1	2	3
4	5	6
7	8	9

Convolution template

Figure 3: Principle of 2D image convolution.

$$\int_z q(z|x)\log\left(\frac{q(z|x)}{p(z|x)}\right)\mathrm{d}z = \mathrm{KL}(q(z|x)\|p(z|x)). \tag{9}$$

The minimum value of (8) is defined as its former part since the value of (9) is greater than 0. The value of $\log\ p(x)$ is fixed when $p(x|z)$ is fixed. Therefore, the value of KL divergence approaches 0 by adjusting $q(z|x)$ and $p(x|z)$. The $\log\ p(x)$ can be equivalent to the former part of equation (8). Then, the following equation can be obtained:

$$L = \int_z q(z|x)\log\left(\frac{p(z,x)}{q(z|x)}\right)\mathrm{d}z. \tag{10}$$

The following result can be obtained by splitting and expressing the divergence of (10):

$$L = -\mathrm{KL}(q(z|x)\|p(z)) + \int_z q(z|x)\log\ p(z|z)\,\mathrm{d}z. \tag{11}$$

The latter part of (11) is image loss function, and the former part can be calculated by its variance and average. Then, the following can be obtained by splitting the former part:

$$\begin{aligned} -KL(q(z|x)\|p(z)) = &\int_z q(z|x)\log\ p(z)\mathrm{d}z \\ &- \int_z q(z|x)\log\ q(z|x)\mathrm{d}z. \end{aligned} \tag{12}$$

$p(z)$ is assumed to obey the distribution of $(0, 1)$, and the following result can be obtained by (12):

$$-KL(q(z|x)\|p(z)) = \frac{1}{2}\sum\left[1 + \log\left(\sigma^2\right) - \mu^2 - \sigma^2\right]. \tag{13}$$

The distribution of $z$is composed of multiple Gaussian distributions. But it is necessary to introduce a penalty coefficient β to (11). This process can reduce the influence of irrelevant factors [24]. Then, the following equation can be obtained:

$$L = -\beta\cdot\mathrm{KL}(q(z|x)\|p(z)) + \int_z q(z|x)\log\ p(x|z)\,\mathrm{d}z. \tag{14}$$

The output vectors obtained by β-VAE were connected to the output vectors obtained by the network 1 in a ratio of 1 : 5. Then, they were input into SVM classifier to complete the classification.

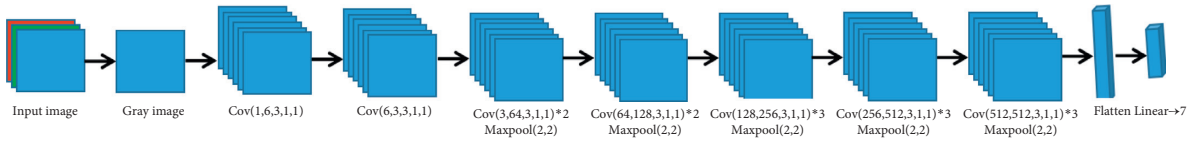

Figure 4: The structure of network 1 composed of 2D convolution.

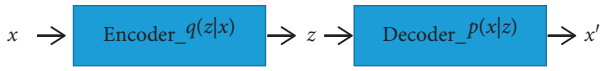

Figure 5: The structure of codec.

2.3. Calculation Parameters Involved in the Model. In this section, some relevant parameters involved in the calculation appeared. Their values are shown in Table 1.

2.4. The Overall Structure of the Model. A combination of feature extraction network and SVM classifier was utilized. Partial facial differences and the influence of noise can be reduced by the facial texture features and the generative features. The overall structure of the proposed model is shown in Figure 6.

3. Results and Discussion

3.1. Preprocessing of Expression Sequence. The CK+ dataset utilized for model training required face alignment and selection of peak image. This process was carried out sequentially since the noise of image background needed to be reduced first to obtain highest testing accuracy. Partial result of face alignment is shown in Figure 7.

The selection of peak images achieved the deletion of edge expressions. Peak images were screened and saved based on Algorithm 1. Partial result of comparison is shown in Figure 8.

The dataset was divided after finishing the preprocessing of it. The ratio of training set, validation set, and testing set was 0.7 : 0.15 : 0.15.

3.2. Neural Network Parameter Settings. The training parameters of network 1 and β-VAE are shown in Tables 2 and 3, respectively.

The task of network 1 focused mainly on the extraction of facial texture features, so the size of input images is [224, 224].

The specific parameters required for β-VAE are shown in Table 3.

The β-VAE took a long time to train as a generative model, so its input size was set to [64, 64]. This model was utilized to extract generative features of image that cannot be captured by network 1.

3.3. Model Evaluation Method. The evaluation of the proposed model in the paper was the accuracy of the testing set, which was calculated by the matching degree between the testing set label and the result of model. The calculation equation of the accuracy was

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}}, \tag{15}$$

where TP refers to the number of target classes predicted as target classes,TN refers to the number of nontarget classes predicted as nontarget classes, FN refers to the number of nontarget classes predicted as target classes, and FP refers to predict the target class as the number of nontarget classes.

Table 1: Convolution network training parameters.

Parameter	Value
Threshold of IoU	[0.6, 0.7, 0.8]
N	11
C_1	$(0.01 \times 255)^2$
C_2	$(0.03 \times 255)^2$
C_3	$1/2\,(0.03 \times 255)^2$
β	2

3.4. Experimental Results. Results of the proposed model were mainly divided into two parts: the feature extraction part and the training part of SVM classifier.

This section described the feature extraction part of the proposed model. The β-VAE introduced by the proposed model consumed many epochs and its loss decreased slowly. The training process is shown in Figure 9.

Intermediate vector extracted by β-VAE was generative, although the process of extraction was slow.

Another component of the proposed model was convolutional network 1. Lenet5 and Vgg19 were utilized as contrast networks to prove the good adaptability of the proposed model. The training process was carried out with the same training parameters. The result of verification is shown in Figure 10.

The verification accuracy of three models on the training set had risen to a high position as shown in Figure 10. The accuracy of the three models on the testing set was 99.422%, 90.366%, and 98.844%, respectively.

Network 1 had the highest accuracy and its rate of accuracy tended to stabilize quickly compared with the other model. But its testing accuracy was lower than the verification accuracy. Network 1 needed to be combined with β-VAE to achieve final effect. The output vectors of above two models were connected by the ratio of 5 : 1. The training of SVM included the selection of kernel function γ and penalty coefficient c. The selection result of the kernel function is shown in Figure 11.

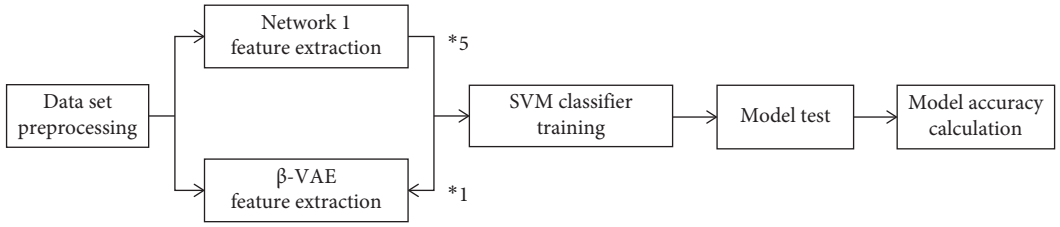

Figure 6: The overall structure of the model.

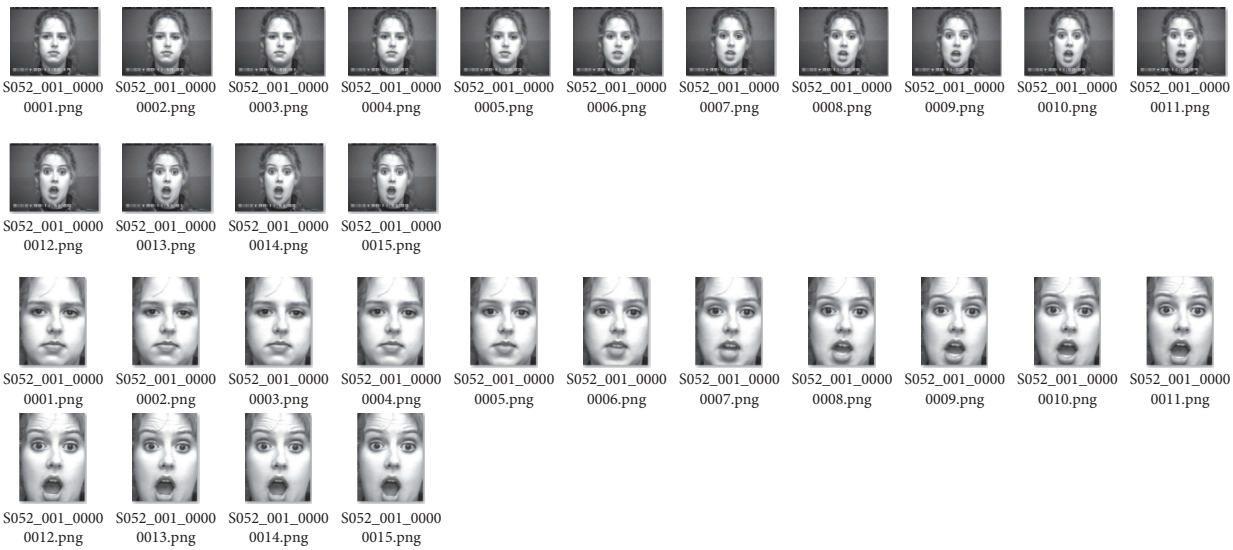

Figure 7: The comparison of face alignment result.

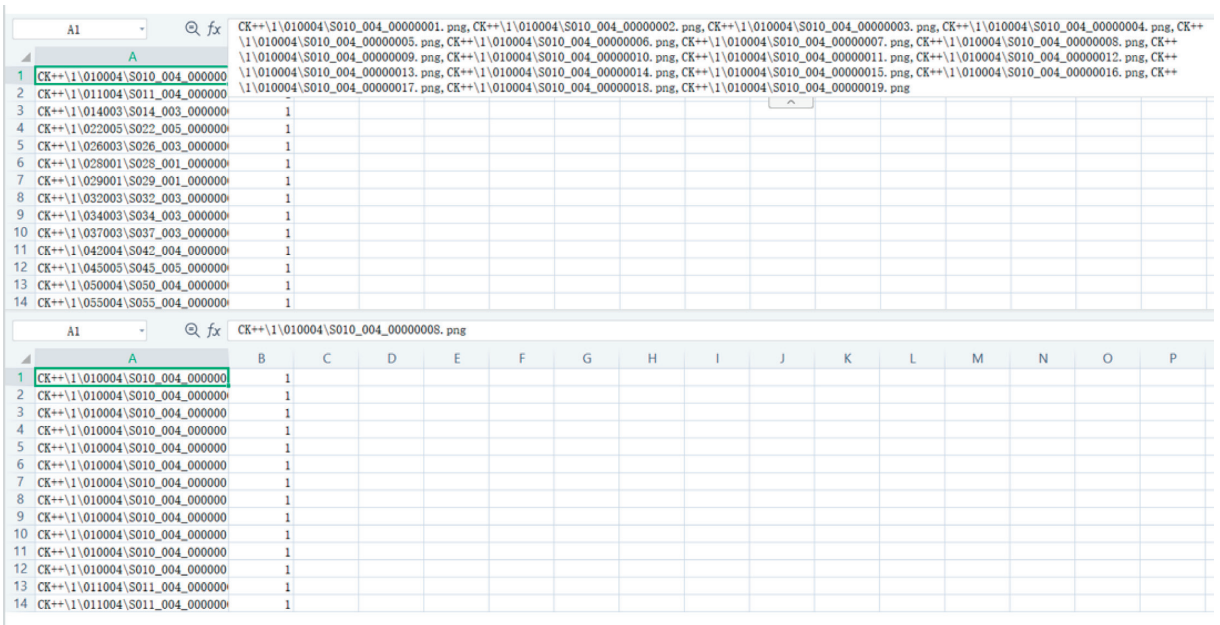

Figure 8: Partial result of comparison.

Table 2: Training parameters of convolution network 1.

Parameter	Value
Batch_size	12
Image_size	[224, 224]
Learning_rate	$1e-3$
Epoch	30

Table 3: Training parameters of β-VAE.

Parameter	Value
Batch_size	32
Image_size	[64, 64]
Learning_rate	$1e-3$
Epoch	200

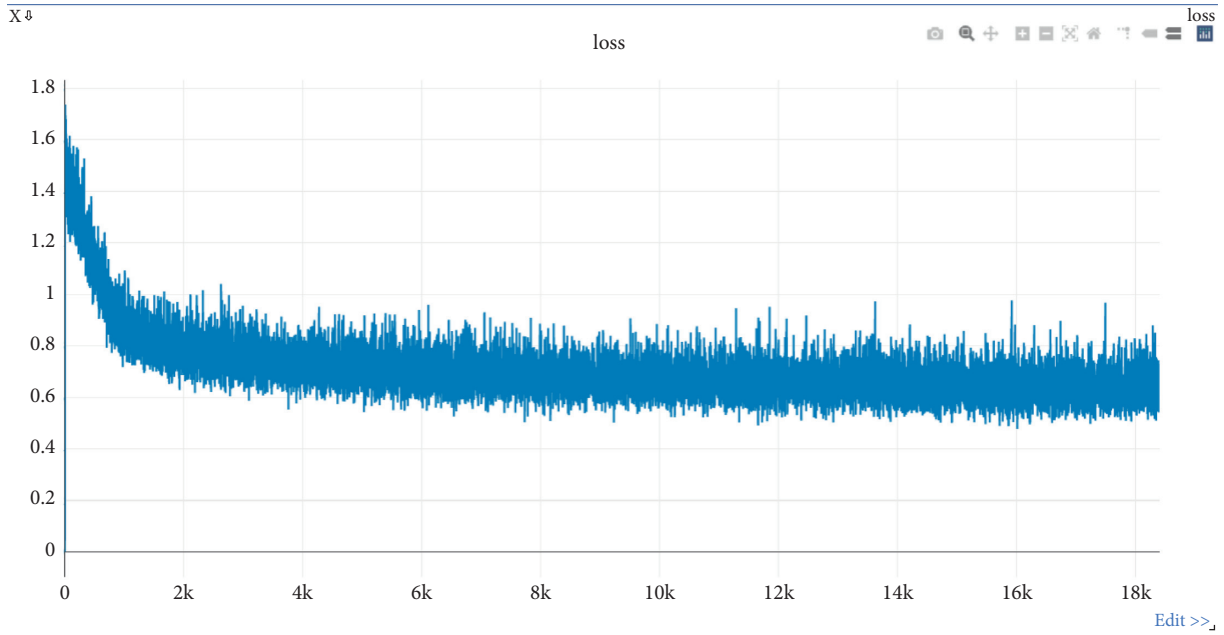

Figure 9: The training process of β-VAE.

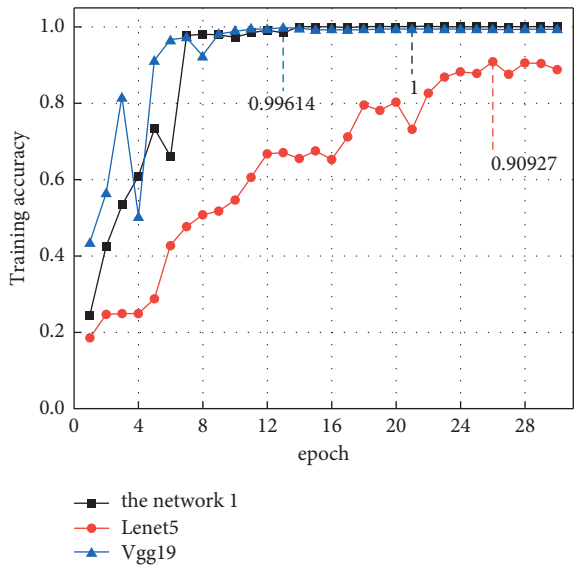

Figure 10: The training process of models.

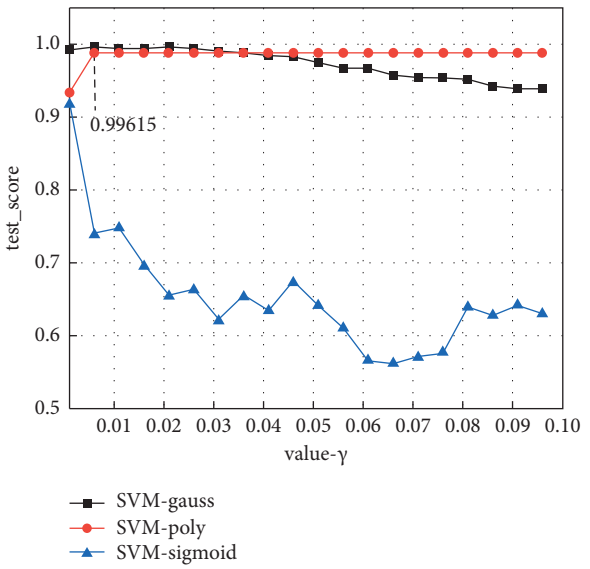

Figure 11: The classification accuracy of SVM classifier on testing set varied with different kernel functions and values of γ.

It can be found that the highest testing accuracy was 99.615% when the kernel function was Gauss, and its value was 0.006. The peak testing accuracy was about 98.844% and 91.715%, respectively, when the kernel function was poly and sigmoid. The best value of γ corresponding to three kernel functions can be determined to be 0.006, 0.006, and 0.001 via fine-tuning. Then, the penalty coefficient c can be adjusted to obtain final classifier. Its process is shown in Figure 12.

It can be found that only sigmoid kernel function was sensitive to changes in value of c. The testing accuracy of the proposed model was 99.615% when its kernel function was Gauss, the value of γ is 0.006, and c was 15.

Final accuracy had a small improvement in recognition compared with the network 1, but it was significantly higher than other comparison models. The above results were obtained from experiments on the preprocessed dataset. The training process on original CK+ dataset is shown in Figure 13.

It can be found from Figure 13 that the performance of models on the original dataset was worse, because their input image had too much interference and noise. Lenet5 has the largest drop, which illustrated the limitation of insufficient depth of convolutional network. The models at this time

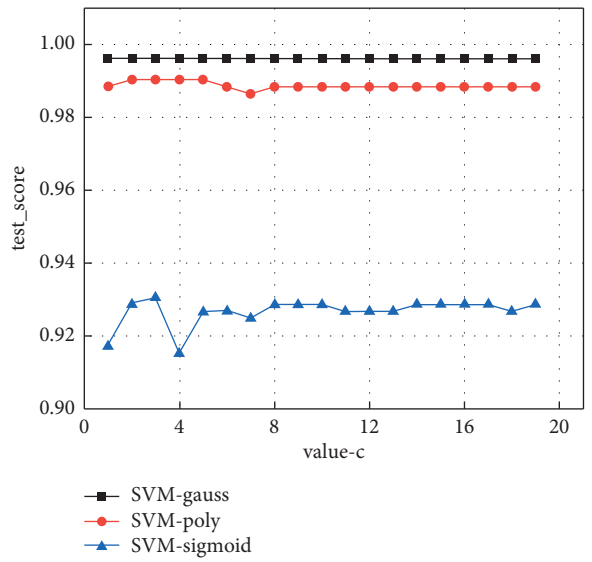

Figure 12: The classification accuracy of SVM classifier on testing set varied with different kernel functions and values of *c*.

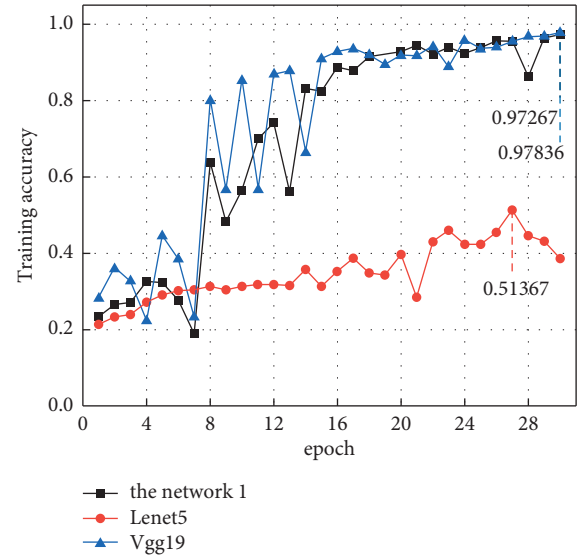

Figure 13: The original CK+ dataset training process result.

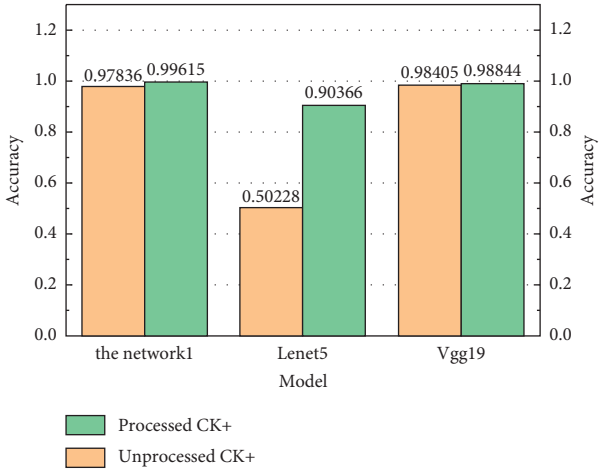

Figure 14: Testing set test accuracy before and after dataset preprocessing.

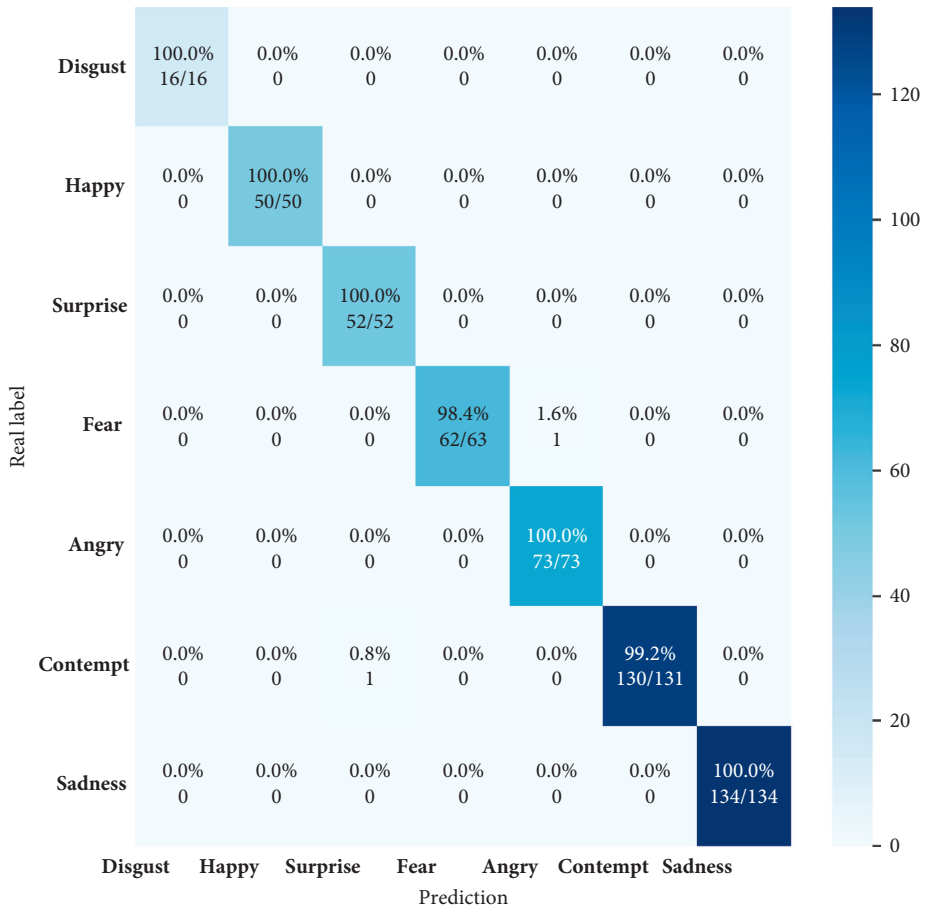

Figure 15: Confusion matrix of classification results of the proposed model.

were tested on the testing set, and the final comparison result is shown in Figure 14.

It can be found from Figure 14 that the highest testing accuracy of the proposed model is 99.615%, which further illustrated the rationality and advancement of it. Confusion matrix was utilized to further display recognition results of the proposed model, which is shown in Figure 15.

The following conclusions can be obtained by analyzing Figure 13:

(1) The quality of input image was high, and there is no error in other facial expression recognition except for fear and contempt

(2) A fear expression had similarity to an angry expression

(3) The result of disgust expression recognition was random due to its small number of samples

4. Conclusions

The facial expression recognition model proposed in the paper can achieve a great testing accuracy. The image preprocessing process made the proposed model converge faster and more stably. It had a significant effect on improving recognition accuracy. In summary, the proposed model had certain advantages and practicability in high-quality frontal expression recognition.

Data Availability

The CK+ data set was used to support this study and its application details are in http://www.jeffcohn.net/wp-content/uploads/2020/04/CK-AgreementForm.pdf. The data set is cited at relevant places within the text as references.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported by Science and Technology Program of Guangzhou (no. 2019050001), Project of Shenzhen Science and Technology Innovation Committee (JCYJ20190809145407809), Project of Shenzhen Institute of Information Technology School-level Innovative Scientific Research Team (TD2020E001), Program for Guangdong Innovative and Entrepreneurial Teams (no. 2019BT02C241), Program for Chang Jiang Scholars and Innovative Research Teams in Universities (no. IRT_17R40), Guangdong Provincial Key Laboratory of Optical Information Materials and Technology (no. 2017B030301007), Guangzhou Key Laboratory of Electronic Paper Displays Materials and Devices (201705030007), and the 111 Project.

References

[1] A. Diete, T. Sztyler, L. Weiland et al., "Recognizing grabbing actions from inertial and video sensor data in a warehouse scenario," *Procedia Computer Science*, vol. 110, pp. 16–23, 2014.

[2] S. Kwon, D. Park, H. Bang, and Y. Park, "Real-time and parallel semantic translation technique for large-scale streaming sensor data in an IoT environment," *Journal of KIISE*, vol. 42, no. 1, pp. 54–67, 2015.

[3] S. Turabzadeh, H. Meng, R. M. Swash, M. Pleva, and J. Juhar, "Facial expression emotion detection for real-time embedded system," *Technologies*, vol. 6, 2018.

[4] D. Mehta, M. Siddiqui, and A. Javaid, "Facial emotion recognition: a survey and real-world user experiences in mixed reality," *Sensors*, vol. 18, no. 2, 416 pages, 2018.

[5] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-PIE," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.

[6] P. Lucey, J. F. Cohn, T. Kanade et al., "The extended Cohn-Kanade dataset (CK+): a complete dataset for action unit and emotion-specified expression," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pp. 94–101, IEEE, San Francisco, CA, USA, June 2010.

[7] Q.-r. Mao, X.-y. Pan, Y.-z. Zhan, and X.-j. Shen, "Using Kinect for real-time emotion recognition via facial expressions," *Frontiers of Information Technology & Electronic Engineering*, vol. 16, no. 4, pp. 272–282, 2015.

[8] H. Y. Yang, U. Ciftci, and L. J. Yin, "Facial expression recognition by de-expression residue learning," *International Journal of Computer Science and Engineering*, vol. 3, no. 2, pp. 2220–2224, 2018.

[9] F. Zhang, T. Zhang, Q. Mao et al., "Joint pose and expression modeling for facial expression recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3359–3368, Salt Lake, UT, USA, June 2018.

[10] C. M. Kuo, S. H. Lai, and M. Sarkis, "A compact deep learning model for robust facial expression recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Salt Lake, UT, USA, June 2018.

[11] Z. Meng, P. Liu, J. Cai et al., "Identity-aware convolutional neural network for facial expression recognition," in *Proceedings of the 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pp. 558–565, Washington, DC, USA, June 2017.

[12] L. T. W. Cheng and J. W. Wang, "Enhancing learning performance through classroom response systems: the effect of knowledge type and social presence," *International Journal of Management in Education*, vol. 17, no. 1, pp. 103–118, 2019.

[13] R. W. Picard, *Affective Computing*, The MIT Press, Cambridge, MA, USA, 2019, https://www.media.mit.edu/groups/affective-computing/overview/.

[14] P. Ekman and W. V. Friesen, "Facial action coding system (FACS): a technique for the measurement of facial actions," *Rivista di Psichiatria*, vol. 12, no. 47, pp. 126–138, 1978.

[15] L. Shan and W. Deng, "Deep facial expression recognition: a survey," *IEEE Transactions on Affective Computing*, no. 99, p. 1, 2018.

[16] E. Yadegaridehkordi, N. F. B. M. Noor, M. N. B. Ayub, H. B. Affal, and N. B. Hussin, "Affective computing in education: a systematic review and future research," *Computers & Education*, vol. 142, Article ID 103649, 2019.

[17] J. Wang, L. J. Yin, X. Z. Wei et al., "3D facial expression recognition based on primitive surface feature distribution," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 1399–1406, New York, NY, USA, June 2006.

[18] W. B. Putra and F. Arifin, "Real-time emotion recognition system to monitor student's mood in a classroom," *Journal of Physics: Conference Series*, vol. 1413, Article ID 012021, 2019.

[19] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[20] B. R. Jiang, R. X. Luo, J. Y. Mao et al., "Acquisition of localization confidence for accurate object detection," 2018, https://arxiv.org/abs/1807.11590.

[21] Y. Y. Hu, S. Yang, W. H. Yang et al., "Towards coding for human and machine vision: scalable face image coding," *IEEE Transactions on Multimedia*, pp. 1–6, 2020.

[22] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[23] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *Clinical Orthopaedics and Related Research*, vol. 10501 page, 2014.

[24] I. Higgins, L. Matthey, A. Pal et al., "Beta-VAE: learning basic visual concepts with a constrained variational framework," 2016, https://openreview.net/forum?id=Sy2fzU9gl.