Hindawi
Scientific Programming
Volume 2022, Article ID 2211468, 11 pages
https://doi.org/10.1155/2022/2211468

Research Article

Analysis and Research on the Characteristics of Modern English Classroom Learners' Concentration Based on Deep Learning

Yu Shen [1,2]

[1]*Huanghe Science and Technology University, Foreign Languages School, Zhengzhou 450000, China*
[2]*University of Málaga, Department of Linguistic, Literature and Translation, Málaga 29000, Spain*

Correspondence should be addressed to Yu Shen; shenyu@hhstu.edu.cn

Received 25 March 2022; Revised 8 April 2022; Accepted 15 April 2022; Published 21 May 2022

Academic Editor: Jie Liu

There are some problems in modern English education, such as difficulties in classroom teaching quality evaluation, lack of objective evaluation basis in teaching process management, and quality monitoring. The development of artificial intelligence technology provides a new idea for classroom teaching evaluation, but the existing classroom evaluation scheme based on artificial intelligence technology has a series of problems such as high system cost, low evaluation accuracy, and incomplete evaluation. In view of the above problems, this paper proposes a solution of English classroom concentration evaluation system based on deep learning. The program studies the evaluation methods of students' class concentration, class activity, and enrichment degree of teaching links, and constructs an information evaluation system of students' learning process and class teaching quality. Based on the edge computing system architecture, a hardware platform with cloud platform AI+ embedded visual edge computing devices managed by an FPGA deep learning accelerated server was built. The design, debugging, and testing of classroom evaluation and student behavior statistics-related functions were completed. This scheme uses edge computing hardware architecture to solve the problem of high system cost. Deep learning technology is used to solve the problem of low accuracy of classroom evaluation. It mainly evaluates the classroom objectively by extracting indicators such as the students' attention in the classroom, and solves the problems of the students' inattentiveness in the classroom. After the test, the classroom evaluation system designed by the paper runs stably and all functions run normally. The test results show that the system can basically meet the requirements of classroom teaching evaluation application.

1. Introduction

In the early stage of college English course learning, the score is often determined by the final exam results, which cannot fully reflect the students' learning situation, is not conducive to finding problems in the teaching process, improving teaching, and is not conducive to mobilizing students' learning enthusiasm [1]. Students often play with mobile phones or sleep in class. With the continuous development of technology, it provides a new idea for intelligent teaching to automatically obtain the attention state of students in English class by using facial expression recognition method. Facial expression recognition, as a noninvasive method, is more suitable for application in actual classroom teaching, which has a huge space to play both online and offline teaching [2].

In 2006, Geoffrey and Ruslan formally proposed the concept of deep learning [3]. A solution to the problem of "gradient disappearance" is proposed, which is to train the algorithm layer by layer by unsupervised learning method and then tune it by supervised back propagation algorithm. In 2012, the deep neural network (DNN) jointly led by the famous Professor Wu of Stanford University and the world's top computer expert Jeff Ade is the amazing achievement in the field of image recognition [4], successfully reducing the error rate from 26% to 15% in ImageNet evaluation. In 2016, Facebook's DeepFace project based on deep learning technology achieved an accuracy rate of more than 97% in face recognition, almost the same as that of human recognition [5]. At present, for face recognition, the face pixel in the image is required to be at least 64×64, preferably 128×128.

In general, face recognition needs to provide a positive face image; for a certain angle of the face image, as long as the demarcated face feature points can be recognized, the face can also be recognized [6].

Based on the above background, by analyzing the main problems existing in the classroom evaluation system at the present stage, this paper puts forward the modern English classroom concentration evaluation system based on deep learning, analyzes the design requirements from the actual needs, puts forward the system design scheme, and expounds the specific design and implementation process in detail.

2. Related Work

2.1. Research Status of Facial Expression Recognition Based on Deep Learning. Different from traditional methods, deep neural networks can automatically obtain required features from a large number of data samples according to specific tasks. In addition, with the support of GPU computing technology, the deep learning model can be designed to be complex enough to contain hundreds of trillions of parameters, learn the semantic information of a different granularity in the samples, successfully avoid the complex operation of manual feature extraction, and achieve great success in the expression recognition task.

Yang et al. proposed a residual expression recognition algorithm that could filter emotion-irrelevant elements in the paper: faces were divided into two categories—faces with expression and neutral faces without expression, and each expression generation model was trained through cGAN network. The model can generate a neutral face for any input face image, and the expression information is stored in the middle level. Then, for neutral faces, the pixel-level or feature-level methods are no longer used to identify expressions, but a new method is used to learn and generate part of the facial expression information left in the middle level of the model, so as to generate the final classifier [7].

Yang designed an expression recognition structure dependent on images and image sequences, and proposed an illumination enhancement strategy, which can reduce the over-fitting phenomenon in model training. Without significantly reducing the recognition performance, the structure has fewer convolution kernels than the general model, which is convenient for deployment in mobile terminals. Moreover, data sets in three different scenarios are collected to verify the validity of the model [8].

Shi et al. believed that there are many mislabeled data in large-scale facial expression data sets, which may make the model overfit and merge and affect the effective features learned by the model. Based on this, a self-cure network (SCN) method is proposed to suppress the uncertainty in facial expression data sets. By extracting image features through CNN, we learned the weight of each image to obtain the importance of samples, arranged the weights in descending order, divided them into two parts of high attention and low attention by calculating the average weight, and re-marked those samples considered to be wrong by estimating the probability. In this way, the sample label error is effectively solved [9].

Yuan et al. studied the application of facial expression recognition in the field environment. Since there are not too many constraints in the field, problems such as unfixed head posture, facial deformation, and motion blur often exist. In this paper, a convolution vision converter method is proposed: firstly, the attentional selection fusion module is used to fuse the feature images generated by the two branches, and the global attention fusion is used to fuse multiple feature images to capture the distinguishing information. The fused features are tiled into the sequence of visual words, and the relationship between the visual words with global self-attention is modeled [10].

2.1.1. Research Status of Facial Expression Recognition in Classroom Environment. In the current classroom teaching scenario, teachers want to obtain students' concentration in class in real time mainly through on-site questioning and visual observation, which requires extra energy and is highly subjective. With the development of facial expression recognition technology, it has become a new trend to use computers to automatically identify students' emotional states in class.

Cheng et al. designed an intelligent teaching model, which consists of four parts: emotion, course, student, and teacher. The system captures students' emotions through visual tracking technology and facial expression recognition technology, and carries out certain emotional feedback behavior [11].

Whitehill et al. studied the measurement of students' participation in class listening in 2014. It is pointed out that the popular methods of participation measurement include self-report, observation list and rating scale, automatic measurement, and automatic recognition based on computer vision. Computer vision analyzes facial cues, body postures, and gestures and provides an unobtrusive way to estimate student engagement [12].

Sun et al. studied the application of emotion recognition in intelligent teaching and proposed that the basic facial expressions defined by Ekman are not suitable for direct application in teaching scenarios, and multiple facial expressions are not highly correlated with students' emotions in class. The "cognitive skill training" experiment is suitable for classroom use. The data sets obtained are analyzed by using boost (BF) algorithm with box filter feature, support vector machine with Gabor feature, and multinomial logistic regression algorithm with expression output, and the model that can automatically identify student participation is obtained [13].

Cui et al. have implemented a student expression analysis system in the classroom environment, which is used to identify three expressions of students in class: ordinary, happy, and confused [14].

Ahuja et al. constructed a three-dimensional learning state space in his research, in which the horizontal coordinate represents the pleasure dimension of learning

emotional state and measures the recognition of learning. The longitudinal axis is the arousal dimension of learning emotional state, which measures learning fatigue, while the vertical axis is the interest dimension of students' emotional state, which measures learning avoidance. At the same time, a total of 9 expressions, including basic expressions, contempt, and confusion defined by Ekman, are introduced, and different weights are set for these expressions according to the measurement standard as the values on the horizontal axis [15].

3. System-Related Technologies and Algorithms

This paper uses Tengine deep learning framework and OpenVINO development tool suite to implement the deep learning algorithm at the edge end and server end, respectively. In the deep learning algorithm, MTCNN is used to achieve face detection and face image interception function, MobileNet-V2 to achieve face recognition function, and VGG-16 to achieve head posture recognition, head recognition, facial expression recognition, and human posture recognition and other classroom behavior recognition functions. This paper mainly introduces the structure of three neural networks, namely, MTCNN, MobileNet-V2, and VGG-16, as well as the deep learning framework Tengine, which is used to run the deep learning algorithm on the edge of the system, OpenVINO, which is used to run the deep learning algorithm on the server.

3.1. Tengine. In order to realize the functions of face detection, face image capture, and classroom behavior recognition at the edge end of the system, the paper adopts EAIDK-610 development board jointly developed by ARM and Open AI Lab as the core processing platform at the edge end of the system [16]. The EAIDK-610 development board is equipped with an embedded AI development platform, including support for heterogeneous computing library HCL, embedded deep learning framework Tengine, and lightweight embedded computer vision acceleration library BladeCV.

Tengine is a lightweight, high-performance, modular embedded deep learning framework. The deep learning framework is optimized for ARM embedded devices, does not rely on third-party libraries, and can be used across platforms. It supports Android and Linux, and supports GPU and DLA as hardware-accelerated heterogeneous computing resources.

Tengine deep learning framework has the following features:

(1) Lightweight tailoring. Tengine relies only on C/C++ libraries; standard edition volume 3M, lite edition volume 300K; original support for ARM CPU.

(2) Running memory optimization. Shared memory technology dramatically reduces memory requirements.

(3) Support model compression and encryption. Tengine customizes model formats, supports model compression to reduce size, supports model encryption and packaging, and protects intellectual property rights.

According to the neural network operators currently supported by Tengine, the deep learning networks currently supported by Tengine include the following:

(1) Squeeze net.mobile net.mobilenet-v2, ResNet50, AlexNet, GoogleNet, ImageNet classification networks, such as Inception-V3, Inception-V4, and VGG-16;

(2) MTCNN;

(3) SSD, YOLO-V2;

(4) MobileNet-SSD;

(5) Faster RCNN;

(6) Lighten CNN;

(7) Network model can be converted into Tengine model based on other deep learning frameworks.

In the deep learning network supported by Tengine, MTCNN, SSD, YOLO-V2, and other networks can realize face detection function. As the real-time requirement of face detection is not very high, the paper adopts MTCNN algorithm with slow execution speed but higher accuracy to realize the face detection function at the edge. For the classroom behavior recognition function of the system, it is necessary to use the classification network to identify and classify all kinds of classroom behaviors. Because it is necessary to conduct real-time classroom behavior recognition for students in the classroom, and there is not much classification of head, facial expression, and human posture in the recognition process, VGG-16 and ResNet50 networks are selected for testing on the premise of ensuring accuracy. After the test, it is found that ResNet50 has a better effect when the number of classification is large. However, in the class behavior recognition of this system, the number of classification of the recognition results of the head, facial expression, and human posture is small, and the recognition results obtained by using VGG-16 network are better. VGG-16 network is finally adopted to realize the algorithm of the class behavior recognition at the edge.

3.2. OpenVINO. In order to realize the face recognition function on the server side of the system, the paper adopts the FPGA accelerated cloud platform launched by Intel as the system server-side processing platform. The platform is loaded with OpenVINO tool suite developed for CPU + FPGA architecture.

OpenVINO (Open Visual Inference & Neural Network Optimization) is a tool suite that can speed up the development of high-performance computer vision and deep learning visual applications, as described in literature [17]. It

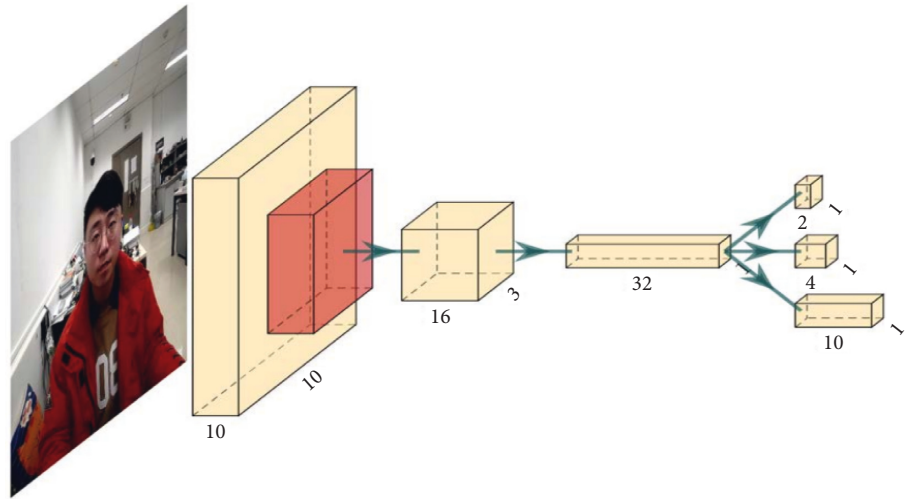

Figure 1: Structure diagram of P-Net.

supports deep learning on various Intel platform hardware accelerators and allows direct heterogeneous execution.

For AI workloads, OpenVINO provides the Deep Learning Deployment Toolkit (DLDT), a suite of tools that can deploy models trained by various open-source frameworks online.

DLDT is divided into two parts:

(1) Model Optimizer. The model optimizer is a Python scripting tool for converting a model trained using an open-source framework into an intermediate representation (IR, intermediate representation) that the inference engine can recognize, that is, two files: the XML file representing the network structure description and the bin file storing the weights, respectively. The model optimizer is used for offline model transformations.

(2) The inference engine is from the inference engine. Reasoning engine is a set of API interfaces supporting C++ and Python, requiring developers to implement their own reasoning process development.

The inference engine is the AI payload deployed to run on the device.

OpenVINO currently supports many network models under mainstream deep learning frameworks. Considering the balance between real-time performance and precision of deep neural network running under embedded hardware, the paper finally uses the improved MobileNet-V2 lightweight network to realize face recognition function on the server side.

3.3. MTCNN. Multi-task convolutional neural network (MTCNN) is a multi-task neural network model proposed by Shenzhen Research Institute of Chinese Academy of Sciences in 2016 for face detection task, as described in literature [18]. The proposed network and refine network output network are used in this model for fast and efficient face detection. The model also uses image pyramid, border regression, and nonmaximum suppression techniques.

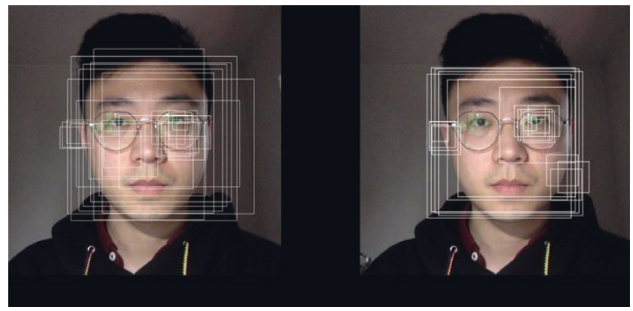

Figure 2: Input and output images of P-Net.

P-Net (proposal network) firstly modified all training samples into $12 \times 12 \times 3$ images, obtained $1 \times 1 \times 32$ feature map (feature map after convolution) through three convolution layers, and finally obtained three multidimensional outputs through three different 1×1 convolution kernels: the $1 \times 1 \times 2$ face probability, $1 \times 1 \times 4$ face candidate box coordinates, and $1 \times 1 \times 10$ face 5 marker anchor point coordinates. Its network structure is shown in Figure 1.

The input and output images of P-Net are shown in Figure 2.

Refine Network (R-Net) modifies all training samples to $24 \times 24 \times 3$ images; after convolution and pooling, the corresponding coordinate positions are output, respectively. Its network structure is shown in Figure 3.

The input and output images of R-Net are shown in Figure 4.

O-Net (output network) modified all training samples into $48 \times 48 \times 3$ images. Like R-Net, the output was face probability, face candidate frame coordinate, and face 5 marker anchor point coordinate, respectively. Its network structure is shown in Figure 5.

The O-Net input and output images are shown in Figure 6.

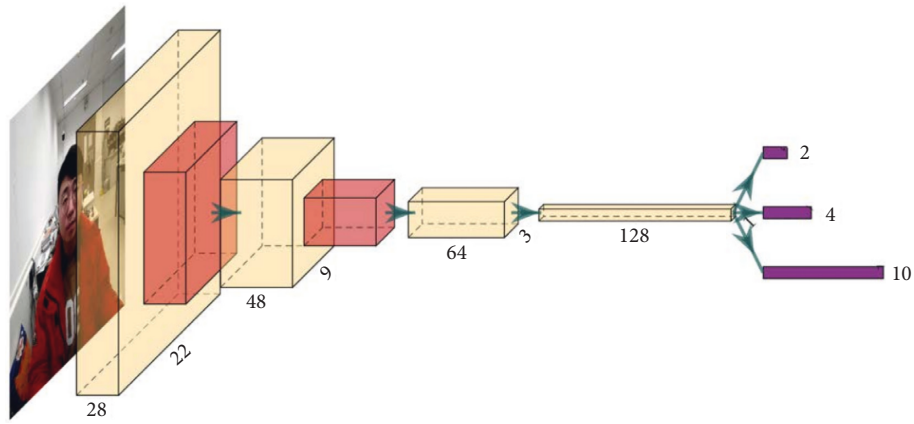

Figure 3: Structure diagram of R-Net.

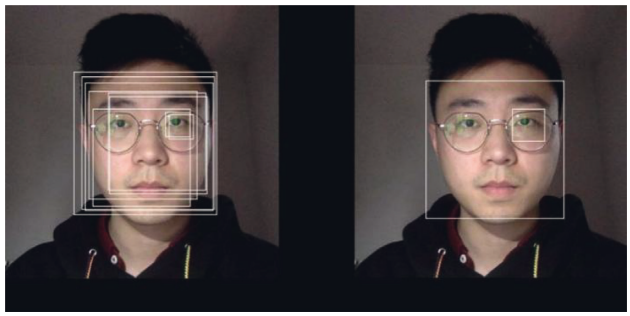

Figure 4: Input and output images of R-Net.

From P-Net to R-Net to O-Net, the larger the size of the input image is, the deeper the network structure is, and the more expressive the extracted features are.

Training MTCNN requires convergence of the following three tasks: face probability, face candidate frame coordinate, and face 5 marker anchor point coordinate.

The following cross drop loss function is used for human face, as shown in

$$L_i^{\text{det}} = -\left(y_i^{\text{det}} \log(p_i) + \left(1 - y_i^{\text{det}}\right)(1 - \log(p_i))\right) y_i^{\text{det}} \in \{0, 1\}. \tag{1}$$

For regression of each candidate box, the following sum of squares loss function is adopted to calculate regression loss through Euclidean distance, as shown in

$$L_i^{\text{box}} = \left\| \hat{y}_i^{\text{box}} - y_i^{\text{box}} \right\|_2^2 y_i^{\text{box}} \in R^4, \tag{2}$$

where $\hat{y}_i^{\text{box}}$ is the coordinate obtained through network prediction and y_i^{box} is the actual real background coordinate, both of which are quaternions.

The following sum of squares loss function is used to calculate the Euclidian distance between the predicted landmark position and the actual real landmark, and the distance is minimized. The calculation formula is shown in

$$L_i^{\text{landmark}} = \left\| \hat{y}_i^{\text{landmark}} - y_i^{\text{landmark}} \right\|_2^2, \quad y_i^{\text{landmark}} \in R^{10}, \tag{3}$$

where $\hat{y}_i^{\text{landmark}}$ is the marking coordinate obtained through network prediction and y_i^{landmark} is the actual real marking coordinate. Since there are 5 marker points for left eye, right eye, nose, left corner of mouth, and right corner of mouth, each point has 2 coordinate values, and y_i^{landmark} is a tuple of ten.

Because each network performs different works, different types of training data are needed during training. The training formula of multiple input sources is shown in

$$\min \sum_{i=1}^{N} \sum_{j \in \{\text{det,box,landmark}\}} \alpha_j \beta_i^j L_i^j, \beta_i^j \in \{0, 1\}. \tag{4}$$

The learning process of the whole training is to minimize the formula above, where N is the number of training samples, α_j is the importance of the task, β_i^j is the sample label, and L_i^j is the loss function mentioned above. When training P-Net and R-Net in MTCNN, $\alpha_{\text{det}} = 1$, $\alpha_{box} = 0.5$, and $\alpha_{\text{landmark}} = 0.5$; when training O-Net, $\alpha_{\text{det}} = 1$, $\alpha_{box} = 0.5$, and $\alpha_{\text{landmark}} = 1$.

3.4. MobileNet-V2. The deep separable convolution is applied in MobileNet-V1, the previous generation of MobileNet-V2, which makes the neural network maintain the accuracy and greatly improve the operation speed. A new structure is proposed in MobileNet-V2 and may be referred to as Inverted Residuals with Linear Bottlelock. This structure first enlarged the dimensions of the input feature map with 1×1 convolution, then operated with 3×3 convolution, and finally reduced the dimensions with 1×1 convolution operation. After the completion of convolution, ReLU activation function is no longer used, but linear activation function is used to retain more feature information and ensure the expressive ability of the model. Its structure is shown in Figure 7.

To improve network performance, the linear convolution part of Inverted Residuals with Linear Bottlelock structure in the network is modified into SE module. Figure 8 shows the structure of an SE block.

Given an input whose channel number is Channel1, a feature whose channel number is Channel2 is obtained through a series of transformations. The feature is input into SE block, and the obtained feature is re-demarcated through three operations:

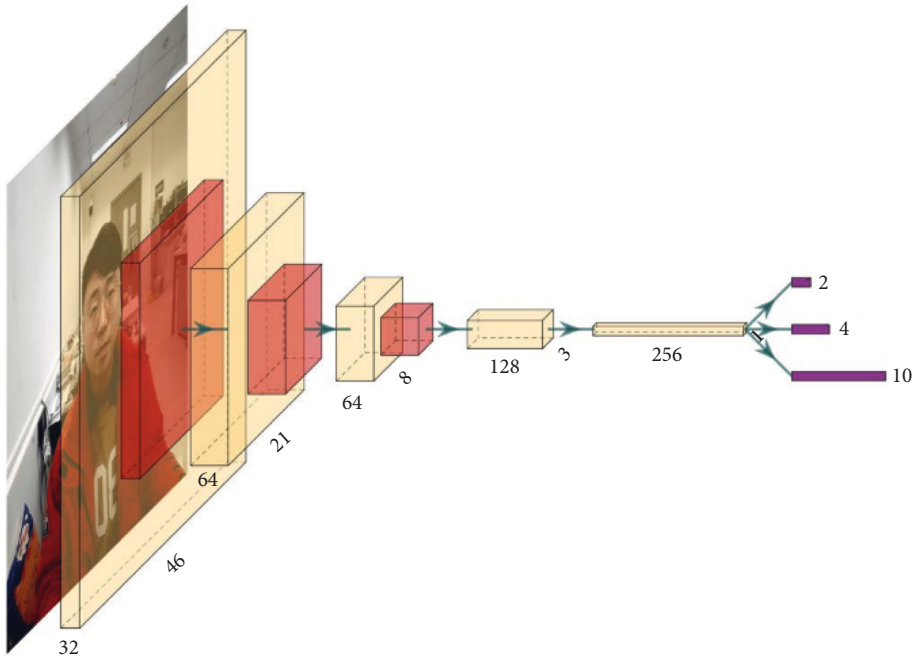

Figure 5: Structure diagram of O-Net.

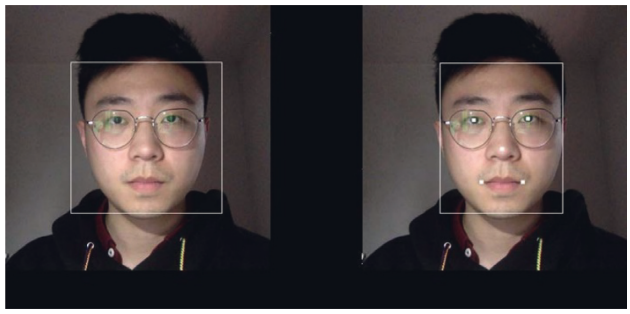

Figure 6: Input and output images of O-Net.

(1) Squeeze operation, use global average pooling to compress features along spatial dimensions, convert each two-dimensional feature channel into a real number, and the output dimension matches the input feature channel number [19]. Its formula is shown in

$$z_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j). \quad (5)$$

(2) The weighting for each attribute channel is generated by the parameter W, where the parameter w is learned to explicitly model the correlation between the attribute channels. Multiply W_1 by the squeeze operation to get z, which is a fully joint operation, and then pass through a ReLU layer with the output dimension unchanged; then, multiply by W_2, which is also a full connection; then finally, through the sigmoid function, you get s. Its formula is shown in

$$s = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W_2 \delta(W_1 z)). \quad (6)$$

The calculation formula of parameter W is shown in

$$W_1 \in \frac{C}{r} \times C, W_2 \in \frac{C}{r} \times C. \quad (7)$$

r is a scaling parameter, whose purpose is to reduce the number of channels and thus reduce the computation. The dimension of the final output is $1 \times 1 \times C$, where C represents the number of channels. s is used to represent the weights of C feature maps.

(3) After obtaining s, use formula (8) for operation [20].

$$\tilde{x}_c = F_{scale}(u_c, s_c) = s_c \cdot u_c, \quad (8)$$

where u_c is a two-dimensional matrix and s_c is the weight, which is equivalent to multiplying every value in the u_c matrix by s_c. The modified Inverted Residuals with Linear Bottlelock structure is shown in Figure 9.

The network inputs a 128×128 image each time, extracts face feature points and feature vectors, respectively, on the image, and finally outputs a row vector containing 256 floating point values, and the row vector is the abscissa and ordinate values of the face feature vector. The cosine distance between the output feature vector and the coordinate value of the feature vector of the reference face image is calculated, and the wide value of the cosine distance is set. If the cosine distance is less than the wide value, the recognized face can be judged as the student himself.

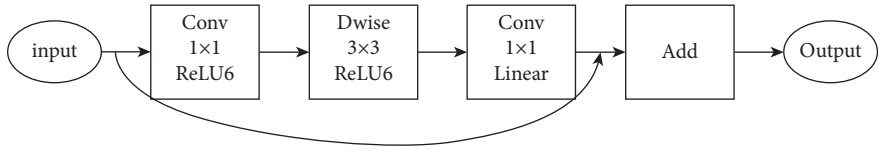

Figure 7: Structure diagram of Inverted Residuals with Linear Bottlelock.

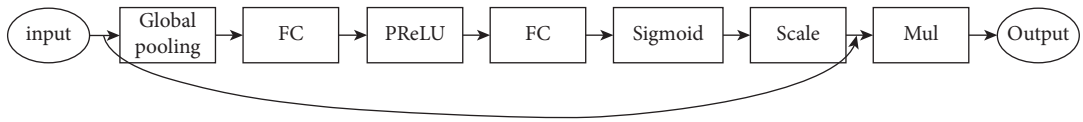

Figure 8: Structure of SE block.

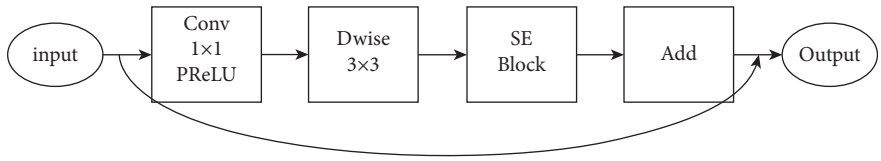

Figure 9: Modified structure of Inverted Residuals with Linear Bottlelock.

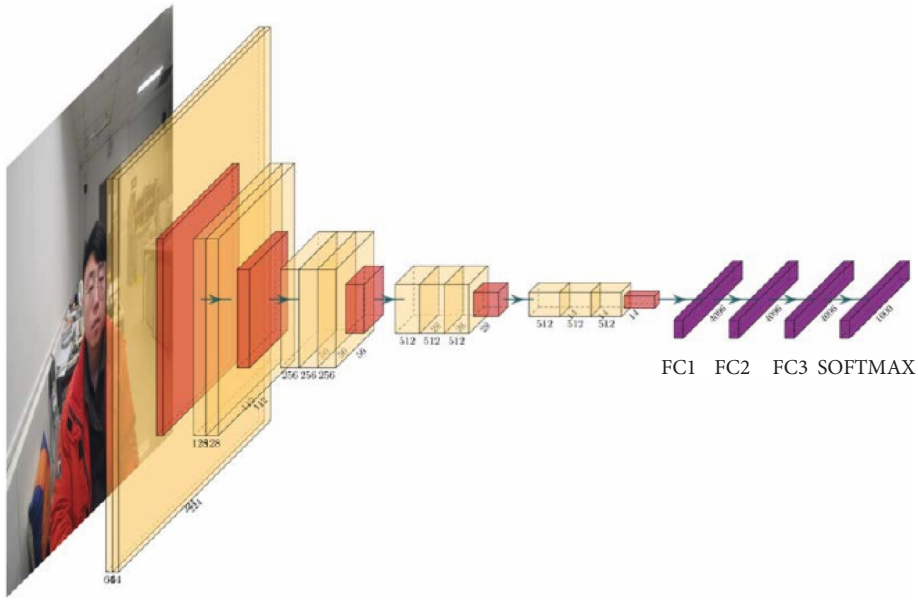

Figure 10: Structure diagram of VGG-16 network.

3.5. VGG-16. VGG is a convolutional neural network model proposed by Simonyan and Zisserman, whose name is derived from the abbreviation of Visual Geometry Group of Oxford University where the authors work [21].

VGG can be divided into six configurations, *A*, *A*-LRN, *B*, *C*, *D*, and *E*, according to the different convolution kernel size and number of convolution layers. Among them, *D* and *E* are more commonly used, called VGG-16 and VGG-19, respectively. Therefore, the system uses VGG-16 architecture.

Figure 10 shows the network structure of VGG-16.

The input data of VGG-16 are $224 \times 224 \times 3$ pixel data. After five-layer convolutional network and pooled network processing, the output is a 4096-dimensional feature data and then processed by three-layer fully connected network, and the final classification result is obtained by Softmax specification. The dimensions of the Softmax layer can be adjusted according to the number of classifications for different purposes.

4. The Realization of the Analysis System for the Characteristics of English Class Students' Concentration

4.1. System Test Environment. In this paper, the classroom evaluation system based on deep learning is completed on

Table 1: Parameters of system edge computing-end hardware test environment.

EAIDK-610 embedded artificial intelligence application development platform	
SoC master chip	RK3399
CPU	2 × Cortex A72 + 4 × Cortex A53
GPU	ARM Mali-T860
Internal storage	4 GB dual-channel 64 bit LPDDR3
Memorizer	16 g eMMC at a high speed
Ethernet	RJ45, 10/100/1000m adaptive

Table 2: Parameters of system server hardware test environment.

Intel FPGA acceleration cloud platform	
CPU	Intel Xeon E5
FPGA	Intel Arria 10 GX
Internal storage	96 G

Table 3: Parameters of system software test environment.

	Testing environment	Versions
The edge of the end	Linux operating system	Fedora 28
Server side	Linux operating system	CentOS 7
Client side	Max operating system	Mac OS 10.15.4

EAIDK-610 embedded artificial intelligence application development platform jointly developed by ARM company and Open AI Lab, and FPGA acceleration cloud platform of Intel Company, combined with webcam and PC. The edge end, the server end, and the client all realize the stable data interaction through the network data transmission module. The hardware test environment parameters of the edge end of the system are shown in Table 1.

Table 2 shows the server hardware test environment parameters of the system.

The software test environment parameters of the system are shown in Table 3.

4.2. System Function Test. According to the analysis of functional requirements, the classroom evaluation system based on deep learning designed in this paper should have the functions of classroom behavior identification, classroom quality evaluation, and student classroom behavior statistics. The following are the functional tests for the above functions [22]. The test was carried out in the laboratory, and the test was set to 10 people, but actually to 9 people. In addition to the two attendance tests, a total of 80 classroom behavior identification test time points were set.

The EAIDK-610 development board is used to control the head and focal length of the camera through the network, so that the camera can aim at the whole student area in the classroom, so that the camera can collect all the students on the seat at the same time and then carry out head posture recognition, head posture recognition, facial expression recognition, and human body posture recognition.

The camera recognizes the head posture of the student in the seat. The system sets a total of 80 test time points for head posture recognition, so the result of head posture recognition after the course has a total of 80 time points of the number of students looking at the platform.

During the same period, the camera looked up at the students in their seats. The system sets a total of 80 test time points for head recognition, so the result of head recognition after the course has a total of 80 time points of head rate data.

During the same period, the cameras recognize the faces of the students in the seats. The system sets a total of 80 test time points for facial expression recognition, so the result of facial expression recognition after the end of the course has a total of 80 time points of the number of students listening carefully.

During the same period, the camera recognizes the posture of the student in the seat. When it is recognized that all students are "sitting down," the class state is defined as "teacher lecturing," and the result value of this state is written as "1." When a student is identified as "standing," the class state is defined as "teacher-student interaction." If the gesture of "raising hands" appears before the gesture of "standing," the result value of the state of "students answering questions independently" is written as "1." If there is no "hand gesture" before the "standing" gesture, the result value of the "teacher calls the roll to answer the question" status is marked as "1." When the student is identified as "writing," the class state is defined as "classroom homework" and the result value of this state is marked as "1."

At the same time of human posture recognition, the system records students' hands raising and sleeping, so there will be data of the number of hands raising and sleeping at the corresponding time point after the course.

Reading face recognition results, head posture recognition results, head posture recognition results, facial expression recognition results, human posture recognition results, and evaluation index calculation according to the class of paper, respectively, calculate the attendance score of the course, head posture correct rate of scoring, serious expression rate hand score, score up rate, rate of score, score not sleep rate, the link of the lecturer score, students' answering question score, teacher roll call to answer questions such as the secondary indicator, class attendance score, class concentration score, class activity score, class link richness score, and other first-level indicators, as well as class evaluation score. Upload the score to the appropriate location on the course evaluation sheet in the database. The database classroom evaluation table is shown in Figure 11.

At the same time, the statistical results of students' classroom behavior are also uploaded to the corresponding position in the statistical table of students' classroom behavior in the database. The statistical table of classroom behavior of students in the database is shown in Figure 12.

After testing, the system designed by the paper has realized the function of classroom evaluation.

4.3. System Performance Test. According to the system design requirements, the factors affecting the system

courseID	A	F1	F2	H	R	J	M1	M2	M3	M4	SA	SF	SV	SM	S
1	90	84.44	63.34	89.96	29.67	89	100	100	0	0	90	80.22	77.71	90	87.79

Figure 11: Classroom evaluation table in database.

courseID	studentID	headupRate	focusRate	raiseRate	sleepRate
1	01	0.938	0.875	0.667	0
1	02	0.975	0.9	1	0
1	03	0.925	0.875	0.333	0
1	04	0.925	0.85	0	0.01
1	05	0.925	0.863	0	0
1	06	0.775	0.725	0	0
1	07	0.925	0.9	0.333	0
1	09	0.787	0.75	0	0.01
1	10	0.912	0.86	0.333	0

Figure 12: Statistics of students' classroom behavior table in database.

Figure 13: Head posture recognition test pictures and test results.

Figure 14: Header recognition test pictures and test results.

Figure 15: Facial expression recognition test results of picture counting test.

performance include face detection rate, face recognition rate, head posture recognition rate, facial expression recognition rate, human posture recognition rate, and system stability.

Figure 16: Human posture recognition test pictures and test results.

Table 4: Technical parameters of system.

Category	Face detection	Face recognition	Head posture	Identify the rise	Identify expression	Recognition of human posture
Test times	50	50	10	10	10	10
Identification of success times	45	47	8	9	8	8
Detection rate	90%	94%	80%	90%	80%	80%

(1) Face detection test

Face detection test is carried out using MTCNN.

(2) Face recognition test

Face recognition is tested using MobileNet-V2 network.

(3) Head posture recognition test

VGG-16 network was used to test head pose recognition.

(4) Header recognition test

The header recognition test is carried out using VGG-16 network.

(5) Facial expression recognition test

VGG-16 network was used to test human pose recognition.

(6) Human posture recognition test

VGG-16 network was used to test human pose recognition.

(7) System stability test

The system stability test tests whether the system is interrupted operation or cannot work normally, the situation. After testing, the system is basically stable.

Test pictures and test results are shown in Figures 13–16.

4.4. Summary of Test Results. Through the function test and performance test of the system, it can be seen that the classroom evaluation system designed in this paper has good human-computer interaction and has realized the functions of automatic classroom attendance, classroom behavior identification, and classroom overall evaluation. The face detection rate, face recognition rate, head posture recognition rate, facial expression recognition rate, and human posture recognition rate of the classroom evaluation system designed in this paper all meet the design requirements. The system stability basically meets the design requirements. The technical parameters of the system are summarized in Table 4.

5. Conclusion

On the basis of functional requirements and design requirements, the paper elaborated the whole development process of classroom evaluation system based on deep learning from the aspects of scheme design, relevant algorithms, detailed design, etc., and carried out functional test and performance test of the designed classroom evaluation system. The test results show that the system has realized the expected function, and all the performance parameters meet the design requirements. The work of the thesis is as follows:

(1) Complete the overall scheme design of the system. According to the research background and application scenarios of the paper, the functional requirements of the system are analyzed to determine the system design requirements, and on this basis, the overall scheme design of the system is proposed, including the hardware platform construction scheme design and software scheme design.

(2) The paper adopts EAIDK-610 embedded artificial intelligence development platform jointly developed by ARM and Open AI Lab as the core processing platform at the edge of the system. On the basis of this platform, the paper realizes the functions of pin-top and focal length control, face detection and interception, classroom behavior recognition, and so on, combined with the spherical webcam. In this paper, Intel FPGA accelerated cloud platform is used as the core processing platform of the system server, and the face recognition, classroom evaluation index calculation, and other functions are realized on this platform.

(3) The paper uses PC as the system client to realize the design of user interaction interface on the client. In addition to the above functions, according to the functional requirements of the system, the paper implements the functions of sending and receiving data transmission on the edge end, server end, and client end, respectively. Finally, the paper realizes the system of classroom behavior recognition of students in fixed positions in the classroom and obtains the function of classroom evaluation according to the above results.

(4) The function test and performance test of the classroom evaluation system based on deep learning designed in this paper are carried out, and the test results are summarized.

Data Availability

The data set can be accessed upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] X. Wang, Y. Zhao, and F. Pourpanah, "Recent advances in deep learning," *International Journal of Machine Learning and Cybernetics*, vol. 11, no. 4, pp. 747–750, 2020.

[2] V. A. Sindagi, V. M. Patel, and M. Vishal, "A survey of recent advances in CNN-based single image crowd counting and density estimation," *Pattern Recognition Letters*, vol. 107, pp. 3–16, 2018.

[3] H. Geoffrey and S. Ruslan, "Reducing the dimensionality of data with neural networks[J]," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[4] A. Brunetti, D. Buongiorno, G. F. Trotta, and V. Bevilacqua, "Computer vision and deep learning techniques for pedestrian detection and tracking: a survey," *Neurocomputing*, vol. 300, no. 15, pp. 17–33, 2018.

[5] 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.

[6] H. Wenlin, "Edge computing enabled non-technical loss fraud detection for big data security analytic in smart grid[J]," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 4, pp. 1697–1708, 2020.

[7] H. Yang, U. Ciftci, and L. Yin, "Facial expression recognition by de-expression residue learning[J]," *International Journal on Computer Science & Engineering*, vol. 3, no. 5, pp. 2220–2224, 2018.

[8] N. N. Yang, "Research on face detection Algorithm based on deep learning[J]," *Science and Technology Innovation Herald*, vol. 15, no. 26, pp. 161-162, 2018.

[9] W. Shi, G. Pallis, and Z. Xu, "Edge computing [scanning the issue]," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1474–1481, 2019.

[10] M. Y. Yuan, C. S. Zhou, H. B. Huang et al., "Review of pooling methods for convolutional neural networks," *Journal of Software Engineering and Applications*, vol. 9, no. 5, p. 13, 2020.

[11] M. M. Cheng, M. S. Lin, and Z. F. Wang, "Research on intelligent teaching system based on expression recognition and sight tracking [J]," *Distance Education in China*, vol. 5, pp. 59–64, 2013.

[12] J. Whitehill, Z. Serpell, Y.-C. Lin, and A. J. R. Foster, "The faces of engagement: automatic recognition of student e facial expressions," *IEEE Transactions on Affective Computing*, vol. 5, no. 1, pp. 86–98, 2014.

[13] B. Sun, Y. N. Liu, J. H. Luo et al., "Expression feature extraction based on tensor analysis [J]," *Computer Engineering and Applications only official website*, vol. 20, pp. 145–148, 2016.

[14] X. K. Cui, T. H. Wang, and Z. P. Zhuang, "Study on emotion recognition technology in student learning process based on OpenCV [J]," *Instrument user*, vol. 25, no. 3, pp. 16–18, 2018.

[15] K. Ahuja, D. Kim, F. Xhakaj, and V. A. S. J. E. C. A. Y. Varga, "EduSense," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 3, pp. 1–26, 2019.

[16] C. Wang and H. Y. Qi, "Visualising the knowledge structure and evolution of wearable device research[J]," *Journal of Medical Engineering & amp; Technology*, vol. 45, no. 3, pp. 112–115, 2021.

[17] Q. Y. Jiang, Y. W. Zhang, S. Q. Tan et al., "Student classroom behavior recognition based on residual network [J]," *Modern Computer*, vol. 20, pp. 23–27, 2019.

[18] D. Kim, C. Park, J. Oh, and H. Yu, "Deep hybrid recommender systems via exploiting document context and statistics of items," *Information Sciences*, vol. 417, pp. 72–87, 2017.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[20] Y. Wang and J. Tang, "Personalized paper recommendation algorithm based on deep learning [J]," *Wireless Communications and Mobile Computing*, vol. 32, no. 4, pp. 35–37, 2018.

[21] L. Guo and L. Liu, "Research on traffic classification method based on multi-layer perceptron [J]," *Journal of Electronic Measurement and Instrument*, vol. 7, pp. 56–64, 2019.

[22] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation networks [J]," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 99, p. 1, 2017.