Journal of Computer Science and Electrical Engineering
ISSN: 2663-1946
DOI: https://doi.org/10.61784/jcsee3044

OPTIMIZATION OF STUDENT CLASSROOM BEHAVIOR RECOGNITION ALGORITHM BASED ON DEEP LEARNING

YunJiao Duan, HaiJun Zhang*
College of Computer Science and Technology, Xinjiang Normal University, Urumqi 830054, Xinjiang, China.
Corresponding Author: HaiJun Zhang, Email: zhjlp@163.com

Abstract: The identification and analysis of student behaviors in the classroom are beneficial for educators to understand and monitor students' learning dynamics and outcomes. Currently, existing deep learning-based classroom behavior recognition models face issues such as low recognition accuracy, limited generalization capabilities, and a narrow dataset coverage, which adversely affect the effectiveness of educational assessments. To address these challenges, this study collected a classroom behavior dataset comprising data from elementary, middle, and high school students and proposed a DCB-YOLOv11 model for student behavior recognition. This model incorporates deformable convolution (DCNv4) in the backbone and detection head of YOLOv8, along with a redesigned CBAM attention module in the pooling layer. The proposed model achieved an average precision of 92.43%, representing a 2.1% improvement over the baseline model, while also significantly reducing computational overhead. This research, combining mobile networks and educational big data, facilitates the personalized development of intelligent learning environments and enhances the effectiveness of the educational process.
Keywords: Deep learning; YOLO; Behavior recognition; Classroom behavior

1 INTRODUCTION

Student behaviors exhibited in the classroom reflect individual learning styles and engagement levels, and they can also indicate the degree of understanding of the knowledge being imparted. Educators can adjust their teaching methods and pacing by understanding students' states. These behaviors are crucial indicators that assessment experts consider during classroom evaluations. In the field of education, commonly used classroom evaluation methods include the Flanders Interaction Analysis System (FIAS), Student-Teacher (S-T) classroom teaching analysis, and Information Technology-based Interaction Analysis Systems (ITIAS). Traditional classroom evaluation methods often require manual observations and detailed on-site recordings. Although manual recording can ensure high-quality qualitative data, it inherently limits the scope for comprehensive quantitative assessments.

With the increasing integration of intelligent technologies in education, the application of deep learning techniques to embed object detection and recognition algorithms into relevant school equipment enables the automatic identification of students' classroom states and behaviors. This approach not only enhances processing speed and recognition accuracy but also broadens the scope of application, which can promote the digital and intelligent development of the education evaluation. In student behavior recognition, object detection methods are primarily categorized into two-stage and single-stage approaches [1].

The two-stage approach is represented by the R-CNN (Regions with Convolutional Neural Networks) series, such as Fast R-CNN and Faster R-CNN. These methods first generate candidate regions and then classify and regress each candidate region. Their advantages are high precision, making them suitable for fine recognition in complex scenarios. However, they have a high computational complexity and slower processing speed, making them unsuitable for applications with stringent real-time requirements. In contrast, the single-stage approach is exemplified by YOLO (You Only Look Once) and SSD (Single Shot MultiBox Detector). YOLO models perform detection and classification directly on the entire image in a single step, significantly increasing detection speed. These methods are computationally efficient and well-suited for real-time detection, but their performance in terms of detail and complexity may not match that of the two-stage approach.

Despite significant advancements in both methods for object detection and behavior recognition, there are still many challenges for recgnitions of classroom behaviors. The two-stage approach, while precise, consumes substantial computational resources, making it difficult to implement on resource-constrained devices. The single-stage approach, although fast, still requires improvement in handling complex scenes and fine detail recognition. Moreover, the generalization ability of existing models across different age groups and classroom environments needs further enhancement to adapt to diverse educational contexts. To address these issues, this paper proposes the DCB-YOLOv11 model (DCNv4+CBAM+YOLOv11), aiming to combine the strengths of both approaches by introducing deformable convolutions and attention mechanisms to enhance the model's recognition accuracy and generalization capability, thereby achieving efficient and precise recognition of student behaviors in the classroom. The main contributions of this study are as follows:

(1) Construction of a Student Classroom Behavior Dataset: This dataset is collected from real classroom scenarios involving primary, secondary, and university students. A total of 5,850 frames and 14,200 behavior annotations were manually extracted from 13 classroom sessions. Student behaviors are categorized into five types: attentive listening, reading and writing, raising hands, standing, and whispering.

(2) Replacement of Key Convolutional Layers: Some convolutional layers in the backbone network of YOLOv11 are replaced with deformable convolution (DCNv4), effectively controlling the model's computational complexity. By introducing dynamic spatial transformations and adaptive weighting, computational cost is reduced while maintaining recognition performance.
(3) Integration of an Attention Module: A redesigned CBAM attention module is incorporated into the SPPF pooling layer to effectively extract and utilize critical information from feature maps, enhancing the model's ability to focus on key behavioral features.

2 RELATED WORK

Deep learning algorithms in the target domain can be broadly categorized into two approaches: two-stage and one-stage methods. The two-stage method, represented by R-CNN, first generates candidate regions and then classifies the extracted samples using a convolutional neural network. In contrast, the one-stage method, exemplified by SSD and the YOLO series, directly extracts features within the network to simultaneously predict the location and category of samples. Similarly, algorithms for student behavior detection can also be classified into these two categories based on this conceptual framework [2].

2.1 Student Behavior Detection Based on Two-Stage Approachs

The two-stage method, also known as region-based object detection, consists of two sequential steps: first, extracting candidate regions of the target objects, and then classifying these regions using a CNN network. The following is a review of representative studies that have applied this method to student classroom behavior detection.
Zaletelj et al. [3] utilized Microsoft Kinect software to collect images and categorized student attention into three levels—low, medium, and high—by integrating facial expressions, eye movements, and body postures. By employing decision trees and the k-nearest neighbors (KNN) algorithm, they achieved an accuracy of 0.753. Zheng et al. [4] proposed an intelligent system for analyzing students' classroom behaviors, capable of detecting three behaviors: raising hands, standing, and sleeping. Their algorithm improved upon the R-CNN-based model by incorporating a scale-aware detection head, enabling the detection of students' postures at varying sizes within the frame.Huang Yongkang et al. [5] introduced an algorithm based on a deep spatiotemporal residual convolutional neural network for real-time student behavior recognition in the classroom. By combining object detection and tracking techniques to extract student images, the model learns each target's spatiotemporal behavioral features through the deep spatiotemporal residual CNN, achieving real-time recognition of multiple student behaviors in classroom scenarios.Lin et al. [6] proposed a network structure based on Feature Pyramid Networks (FPN), leveraging multi-scale feature extraction through deep convolutional networks and the hierarchical structure of FPN to efficiently detect objects of different scales. Their method achieved state-of-the-art single-model performance on the COCO detection benchmark within a standard Faster R-CNN system, offering a practical and accurate multi-scale object detection solution.Zhao et al. [7] further improved the FPN network by enhancing the Pyramid mechanism through effective fusion of multi-level features extracted by the backbone. By integrating the Multi-Level Feature Pyramid Network (MLFPN) into the end-to-end single-stage object detector M2Det, their approach surpassed the performance of state-of-the-art single-stage detectors on the MS-COCO benchmark.
Although the two-stage method offers a significant advantage in accuracy, its application in evaluating students' classroom behavior in educational settings is hindered by its slower processing speed and complex network structure. The limitations in real-time performance and computational complexity restrict the practical implementation of two-stage algorithms in the education domain.

2.2 Student Behavior Detection Based on Single-Stage Approachs

Unlike the two-stage method, the one-stage algorithm performs object localization and classification in a single step. It typically employs a dense prediction grid, where each grid cell predicts multiple bounding boxes along with their corresponding class probabilities, enabling rapid object detection within an image. Due to its high speed and real-time performance, the one-stage algorithm, represented by the YOLO series, has made significant advancements in student behavior recognition.
Wang et al. [8] utilized the backbone convolutional layers of YOLOv5 to extract features from input images and incorporated a Squeeze-and-Excitation (SE) attention mechanism to reduce excessive focus on irrelevant information, thereby mitigating background interference.Guo Junqi et al. [9] proposed an improved network structure and loss function for the YOLOv5 model based on the characteristics of classroom scenarios. The model, designed for multi-object detection, was specifically applied to student behavior recognition, and its effectiveness was validated through comparative experiments.Z. Zheng et al. [10] retained the FPN+PAN feature extraction framework while optimizing the CBL modules in YOLOv5 by replacing the default LeakyReLU activation function with the GELU activation function. Additionally, the SIoU loss function was introduced to accelerate convergence. Experimental results demonstrated improvements in both accuracy and detection speed.W. Niu et al. [11] integrated a coordinate attention mechanism into each CSP module within the YOLOv5 architecture to address the issue of missed detections. By decomposing channel attention into one-dimensional features for encoding, the mechanism enhanced output precision.Z. Zhang et al. [12] introduced the CloU loss function to replace the default loss function, addressing the issue of distance

miscalculation when two predicted bounding boxes do not intersect. Additionally, the default activation function was replaced, making the improved YOLOv5-based model better suited for capturing student behavior data features.Yang et al. [13,14] integrated the dual-attention module and Wise-IoU into the YOLOv7 network for classroom behavior detection. Through experiments on their own dataset, they achieved a 1.8-point increase in average precision. Later, they further improved the detection accuracy by incorporating results from YOLOv7 CrowdHuman, slow-fast, and deep sorting models.YOLOv8 [22], a relatively newer version in the YOLO series, introduced further optimizations over YOLOv7 in terms of accuracy, speed, model scalability, and flexibility, enhancing its performance across various application scenarios. It demonstrates particularly improved performance in handling complex scenes and multi-scale objects. YOLOv8 introduces a more robust network architecture that combines the advantages of deep convolutional neural networks and lightweight models, increasing accuracy while ensuring efficiency. The inclusion of an adaptive anchor box generation algorithm allows the model to dynamically adjust the size and position of anchor boxes, further enhancing its performance in small object detection and high-density scenes.

The improvements made to YOLO based on one-stage algorithms are more aligned with the requirements for student behavior recognition. Researchers have continuously refined the algorithm to enhance recognition accuracy while ensuring real-time performance. YOLOv11 [23] is the latest version in the YOLO series. Compared to YOLOv8, the previous CF2 module has been upgraded to C3K2, and a new module, C2PSA, has been added after the SPPF module. Additionally, the YOLOv10 head concept has been incorporated, employing depthwise separable convolutions to improve computational efficiency and reduce redundancy.

In summary, there are several key challenges currently faced in the field of student behavior recognition: first, in classroom environments, students' actions continuously change in response to the teacher's lecture content, and improvements in behavior action accuracy are still significantly limited; second, due to the large number of students in the classroom, the ability to simultaneously recognize multiple behaviors is insufficient; third, the existing datasets lack diversity and coverage, which limits the model's generalization ability and its capacity to adapt to various educational scenarios. These issues collectively affect the reliability and effectiveness of algorithms in practical teaching evaluation.

This paper proposes replacing key convolutional layers in YOLOv11 with DCNv4 and introducing an improved CBAM attention module to enhance the model's ability to represent and accurately detect student behavior features. This combination aims to maintain high efficiency while improving the model's recognition performance in complex scenarios, addressing the dual requirements of real-time performance and accuracy. Additionally, by expanding and optimizing the existing dataset, a more comprehensive coverage of various student behaviors and classroom environments will be achieved, further enhancing the model's generalization ability and adaptability.

3 METHOD AND MODEL DESIGN

3.1 Model Overview

In classroom environments, there are often a large number of students, and the relative sizes of individuals in the same frame vary, which raises the demands on object detection algorithms. This paper uses YOLOv11 as the backbone network and replaces the convolutional layers in both the Head and Backbone sections with DCNv4 convolutions. In this process, the kernel size, stride, and padding method of the new convolutions are kept consistent with the original layers to ensure that the feature map dimensions remain unchanged. Additionally, an improved CBAM attention module is integrated into the pooling layer. The overall model structure is shown in Figure 1.

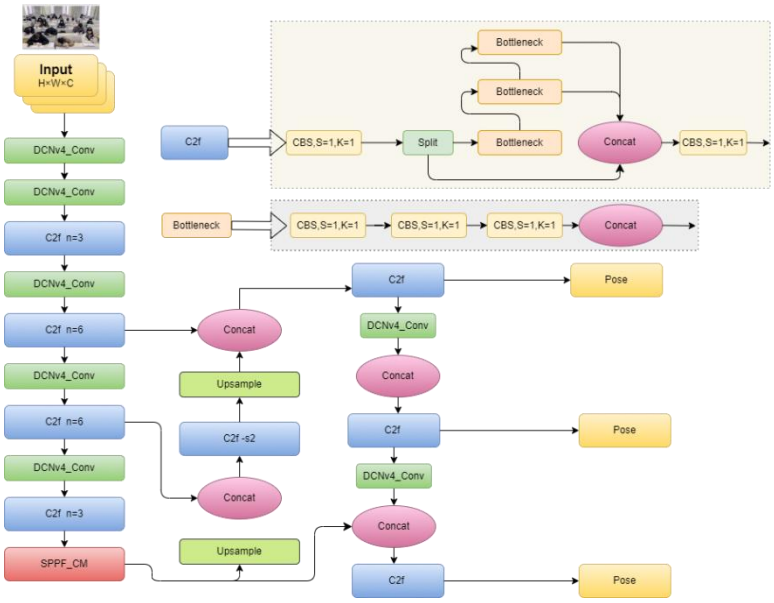

Figure 1 Structure of the DCB-YOLOv8 Model

DCNv4 is an enhanced convolutional network that adaptively adjusts the position of convolutional kernels, allowing for better capture of the deformation features of objects. This adaptability results in superior performance when handling targets of varying sizes. Additionally, the incorporation of the CBAM attention mechanism based on multi-scale feature fusion in the pooling layer enhances the expressive power of the features. This approach not only utilizes multi-scale information but also further optimizes the representation of feature maps through the CBAM mechanism, which improves model performance across various tasks. By integrating DCNv4 and the improved CBAM attention module, the model achieves further optimization in feature extraction and representation which can enhance the robustness and accuracy of object detection, particularly in complex classroom scenarios with many students.

3.2 Replacement of Standard Convolutions with DCNv4 Deformable Convolutions

Although YOLOv11 incorporates a refined architecture design and an optimized training process, achieving faster processing speed with an effective balance between accuracy and performance, there is still room for improvement in recognizing specific scenarios in this study. The default convolutions in YOLOv11 tend to overlook important details when recognizing student behaviors of varying sizes in the classroom, which may lead to misclassifications. Replacing certain convolutional layers in YOLOv11 is thus essential for the accurate behavior recognition in the specific classroom scenarios addressed in this paper.

In this paper, the YOLOv11 model is improved by replacing the conventional convolutional layers in its Head and Backbone with a Deformable Convolutional Network (DCNv4) [15]. DCN is a deformable convolutional neural network module used for object detection and image segmentation. The DCN series significantly enhances convolution adaptability by adding learnable offsets to the convolutional kernels. Unlike standard convolutions, attention mechanisms, due to their ability to model long-range dependencies, have been successfully applied to various computer vision tasks. Window attention limits the attention operation to a fixed-size window, thereby reducing the computational complexity of regular attention. To further reduce the high computational complexity introduced by standard attention, deformable attention assigns dynamic positions and weights, allowing each query to focus on a specific number of key sampling points.

DCNv4 introduces deformable convolutional kernels that can adaptively adjust their shapes, enabling a more effective capture of spatial deformation features. The standard convolution operation can be expressed as follows:

$$y(p_0) = \sum_{p_n \varepsilon R} \omega(p_n) \cdot x(p_0 + p_n) \quad (1)$$

In this context, $y(p_0)$ represents the value of the output feature map at position p_0, and x denotes the input feature map, and ω refers to the convolutional kernel weights, and p_n is the set of sampling positions.

In DCNv4, deformable Δp_n offsets are introduced, allowing the convolution operation to adaptively select sampling positions. The specific formula can be changed as follows:

$$y(p_0) = \sum_{p_n \varepsilon R} \omega(p_n) \cdot x(p_0 + p_n + \Delta p_n) \quad (2)$$

In the initial module design of DCNv3, the calculation of offsets and dynamic weight allocation was performed by a complex sub-network comprising deep 3×3 convolutions and layer normalization and GELU activation, and linear layers, as shown in Figure 2. In contrast, DCNv4 follows the design principles of Xception by eliminating the normalization and GELU layers while incorporating the original separable convolution structure. This architectural change significantly reduces the computation time required for operations [16].

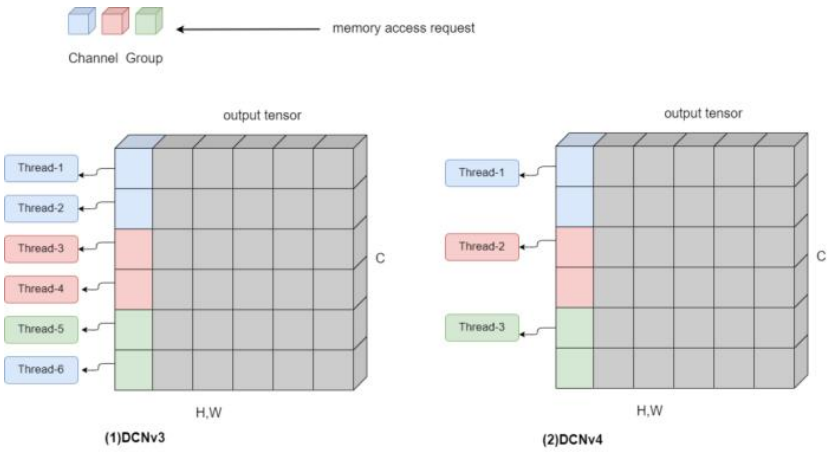

Figure 2 Structure of the DCNv4 Model

In the Backbone of YOLOv11, the convolutional layers are responsible for feature extraction. However, standard convolutional layers are not effective in capturing complex spatial deformations or adaptively adjusting the sampling locations. To address these limitations, the following improvements have been made in this paper.

The first step involves locating all the standard convolutional layers and introducing an offset learning module within these layers. This module is a small convolutional network that takes the feature map as input and outputs the offsets. By replacing the standard convolution operations with deformable convolutions, the Backbone can adaptively sample

the input features, better capturing complex spatial deformations and thereby enhancing the feature extraction capabilities.

In the Head section, where convolutional layers are responsible for object classification and bounding box regression, the second step similarly locates all standard convolutional layers and introduces the offset learning module. This allows the convolution operations in the Head to adaptively adjust the sampling positions, improving the accuracy of classification and bounding box regression.

This replacement strategy enables YOLOv11 to adaptively adjust the shape of the convolutional kernels in both feature extraction and object detection processes, effectively handling complex situations such as deformations, rotations, and scale variations. Experimental results show that this improvement significantly enhances the model's average precision while reducing computational time.

3.3 The Improved SPPF Layer

The original SPPF module in YOLOv11 primarily relies on multi-scale pooling for feature fusion but lacks a refined attention mechanism for feature selection. Additionally, it does not fully utilize global contextual information, which particularly limits its performance in fine-grained object detection and recognition tasks. This is detrimental to behavior recognition in complex classroom environments. In this study, we introduce a CBAM attention mechanism based on multi-scale feature fusion within the SPPF, which enhances the model's understanding and detection capabilities and maximizes the utilization of multi-scale features, and improves overall performance.

CBAM (Convolutional Block Attention Module) [17] is a lightweight attention mechanism for Convolutional Neural Networks (CNNs), primarily consisting of two parts: the Channel Attention Module and the Spatial Attention Module. These two modules separately apply attention to the channel and spatial dimensions. However, this structure overlooks the interaction between the channel and spatial dimensions. Additionally, CBAM lacks a comprehensive consideration of global contextual information, focusing primarily on local features. In the context of classroom environments, global information—such as student distribution and overall classroom atmosphere—is crucial for accurately understanding student behavior. The absence of a global perspective can impact the model's overall judgment ability [18].

Multi-Scale Feature Fusion is a method that enhances feature representation by combining information from different scales. It helps the model capture details and contextual information at various levels, improving overall performance. In this paper, the model's overall performance is enhanced by enriching feature representations and strengthening the interrelationship between features. The specific approach is to implement multi-scale feature fusion, which aids in capturing information about students at different scales. The modified module is named m_CBAM, and the detailed approach is outlined as follows.

Multi-scale features are extracted from the input feature map, denoted as $F_1 \in R^{C\times H\times W}$, where C is the number of channels, and H and W are the height and width, respectively. By applying convolutional kernels of different sizes, three distinct scale feature maps can be obtained:

$$F_1 = \mathrm{Conv}_{1\times1}(F) \quad (3)$$

$$F_3 = \mathrm{Conv}_{3\times3}(F) \quad (4)$$

$$F_5 = \mathrm{Conv}_{5\times5}(F) \quad (5)$$

Here, the $\mathrm{Conv}_{k\times k}$ convolutional kernel sizes are represented as $k \times k$, and next, the multi-scale feature maps are fused by concatenation to obtain a feature fusion map F_{fused}. The symbol "Concat" refers to the concatenation operation along the channel dimension.

$$Ffused = \mathrm{Concat}(F_1F_3,F_5) \quad (6)$$

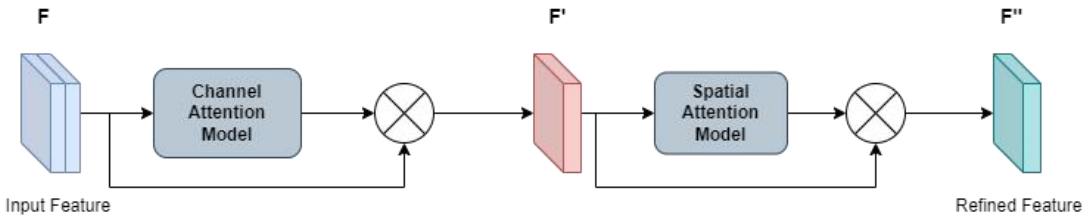

Figure 3 CBAM Model Architectur

As shown in Figure 3, CBAM takes the intermediate feature map F as input and first applies a one-dimensional convolution within the Channel Attention Module. The convolution result is then multiplied by the original feature map to generate the input for the CAM module. Subsequently, the Spatial Attention Module applies a two-dimensional convolution, and the output is multiplied with the original feature map. This process allows the model to focus on the most relevant features, enhancing its ability to interpret and analyze the input effectively [19].

The fused feature map from the Channel Attention Module undergoes global average pooling and global maximum pooling operations, yielding two feature vectors F_{avg} and F_{max}, the two vectors are processed through a shared multi-layer perceptron (MLP) and then summed together. This combined output is passed through a sigmoid function to obtain the channel attention map:

$$F_{avg} = \mathrm{GlobalAvgPool}(F_{fused}) \quad (7)$$

$$F_{max} = \text{GlobalMaxPool}(F_{fused}) \quad (8)$$

$$M_c = \sigma(\text{MLP}(F_{avg}) + \text{MLP}(F_{max})) \quad (9)$$

The Spatial Attention Module performs a similar operation on the fused feature map by applying average pooling and maximum pooling, resulting in two single-channel feature maps F_{avg_s} and F_{max_s}, The two single-channel feature maps are concatenated and passed through a 7×7times convolutional layer. The output is then processed through a sigmoid function to obtain the spatial attention map M_s.

$$F_{avg\_s} = \text{Mean}(F_{fused}, \text{dim} = 1) \quad (10)$$

$$F_{max\_s} = \text{Max}(F_{fused}, \text{dim} = 1) \quad (11)$$

$$M_s = \sigma(\text{Conv}_{7\times7}(\text{Concat}(F_{avg\_s}, F_{max\_s}))) \quad (12)$$

Finally, the channel attention map and the spatial attention map are applied to the fused feature map through element-wise multiplication, represented as:

$$F_{enhanced} = F_{fused} \cdot M_c \cdot M_s \quad (13)$$

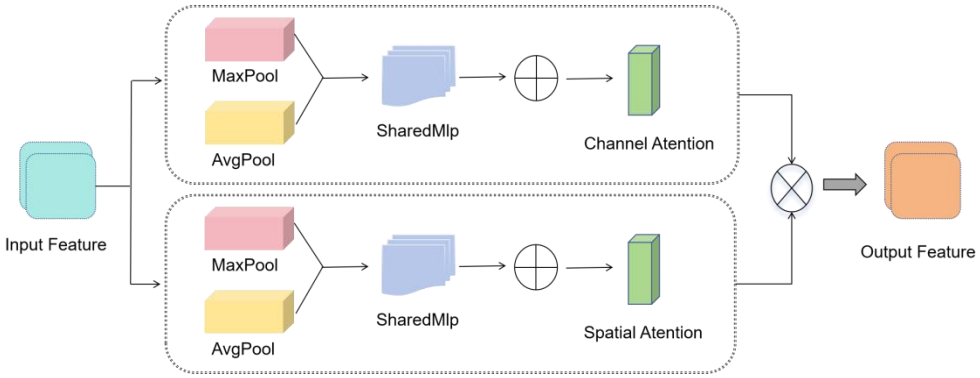

Figure 4 m_CBAM Model Architectur

Through the steps outlined above in Figure 4, the multi-scale feature fusion-based m_CBAM attention mechanism is implemented, enhancing the feature representation capability. This approach effectively leverages multi-scale information and further optimizes the feature map representation through the CBAM attention mechanism, resulting in improved performance of the model across various tasks.

4 EXPERIMENTS DESIGN AND RESULTS ANALYSIS

4.1 Dataset Preparation and Parameter Settings

The study utilized our constructed dataset comprising real classroom scenes from primary, middle, and high school students, ensuring diversity and authenticity in the data. A total of 5,850 video clips and 14,200 behavior annotations were manually extracted from 13 classes. The observed student behaviors were categorized into five types: attentive listening, reading and writing, raising hands, leaning on the desk, and whispering [20]. The dataset was then divided into training and testing sets in a 4:1 ratio to facilitate model evaluation. An illustration of the sample data collection is provided in the figure 5 and 6 below.

Figure 5 Sample Data Collection from the Dataset (University Classroom)

Figure 6 Sample Data Collection from the Dataset (Middle School Classroom)

The analysis of the dataset revealed a significant class imbalance in student behavior categorization. Behaviors such as attentive listening and reading and writing were predominant, while the other three behaviors were relatively infrequent,which can lead to biased model training . To tackle this class imbalance, the study introduced a weighted cross-entropy loss function, where the weight for each class was determined based on its sample proportion relative to the total number of samples in the training set. Additionally, to prevent the underrepresentation of minority classes during training, a customized sampling strategy was implemented. This strategy ensured that samples from all classes were included in the mini-batches during training with Stochastic Gradient Descent (SGD), thus improving the model's robustness.

For experiments, the input image size was set to 640×640 pixels, with a batch size of 16 and an initial learning rate of 0.01, utilizing a cosine annealing learning rate scheduler over 200 batches to achieve smooth learning rate decay. The AdamW optimizer was selected for its effective balance between convergence speed and stability, with a weight decay of 0.005 and a momentum of 0.9. The number of deformable groups in DCNv4 was set to 1, with the modulation mechanism enabled to enhance flexibility in feature extraction.

The data augmentation techniques, including random cropping, flipping, and color jittering, were applied to increase the diversity of the training data and to improve the model's generalization capability. The loss function used in the model comprised localization loss, classification loss, and confidence loss to optimize different aspects of detection performance.

The model training was conducted on a hardware environment featuring a 2.50 GHz, 11th Gen Intel(R) Core(TM) i7-12700H CPU (20 cores) and an NVIDIA GeForce RTX 3090 GPU with 12 GB of VRAM, providing the computational power necessary for efficient training.

4.2 Testing Metrics

In experiments, Recall and Mean Average Precision (mAP) were employed as key metrics for evaluating algorithm performance, alongside considerations of parameter count and Floating Point Operations per Second (FLOP). Recall measures the model's ability to identify all relevant instances, while mAP evaluates detection accuracy across a range of thresholds, providing a comprehensive assessment of the model's precision [21].

Recall evaluates the model's ability to identify all positive samples by calculating the proportion of correctly detected positive samples out of all actual positive samples. Here, TP (True Positive) represents the number of correctly detected positive samples, and FN (False Negative) is the number of positive samples that were missed.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{14}$$

Mean Average Precision (mAP) is calculated as the mean of average precisions (AP) for all categories. Here, N denotes the total number of categories, and APi represents the average precision for the i-th category. The average precision for each category is computed by calculating the area under the precision-recall curve at various thresholds, offering insights into the model's performance across different levels of confidence.

$$\text{mPA} = \frac{1}{\text{N}}\sum_{\text{i=1}}^{\text{N}} \text{AP}_\text{i} \tag{15}$$

The number of parameters reflects the complexity and computational demands of the model, representing the total count of all trainable parameters. Meanwhile, the number of floating point operations per second (FLOP) measures the computational complexity of the model, indicating the number of floating point operations required for a single forward pass. This includes the count of Multiply-Accumulate Operations (MAC). The specific calculation depends on the operations involved in each layer of the model.

$$\text{FLOP} = 2 \times (\text{MAC}) \tag{16}$$

By combining these metrics, we can comprehensively evaluate the model's performance, complexity, and computational requirements, facilitating informed decisions and regarding optimization and selection.

4.3 Comparative Experiments

To validate the accuracy of the improved algorithm, experiments were conducted comparing it with several advanced

algorithms currently used in behavior recognition: YOLOv5, YOLOv8, YOLOv10, and YOLOv11.

Table 1 Recognition Results of Student Classroom Behaviors by Different Algorithm Models

Classroom Behaviors	YOLOv5	YOLOv8	YOLOv10	YOLOv11	**DCB-YOLOv11**
Listening Attentively	0.80	0.72	0.79	0.81	**0.83**
Raising Hands	0.85	0.85	0.94	0.88	**0.94**
Leaning on Desk	0.78	0.82	0.86	0.83	**0.83**
Reading/ Writing	0.81	0.77	0.82	0.81	**0.84**
Whispering	0.74	0.70	0.76	0.77	**0.78**

The four models mentioned above represent the state-of-the-art models in the field of object detection, each with its own strengths and weaknesses, and are commonly chosen for comparison in many studies and experiments. According to the experimental results presented in this paper, the average accuracy of the five algorithms on the same dataset is shown in Table 1 The DCB-YOLOv11 model proposed in this paper outperforms the other models in terms of recognizing most classroom behaviors, except for the behavior of "leaning on the desk," where it does not show a clear advantage over other models.

Upon analysis, it is noted that the "raising hand" gesture exhibits a larger range of motion compared to other classroom behaviors, leading to higher recognition accuracy across all algorithms, with little difference in their performance. For the "whispering" behavior, the recognition accuracy is generally lower due to the camera's positioning, which often misidentifies students sitting at the edge as engaging in whispering, even when they are simply lifting their heads to listen. This issue leads to a lower overall recognition accuracy for this category.

Table 2 Presents the comparative results of the experiments

Algorithm Model	Number of Parameters	Inference Time (ms)	mPA (%)	FLOP(GB)
YOlOv5	1,756729	46	83.97	15.4
YOLOv8	2,946151	65	80.99	13.8
YOLOv10	3,006623	59	84.02	14.2
YOLOv11	2,533114	54	89.50	16.7
DCB-YOLOv11	**2,736232**	**57**	**90.63**	**15.6**

As shown in Table 2, a comprehensive comparison experiment was conducted on advanced models such as YOLOv5, YOLOv8, YOLOv10, and others, evaluating their performance in four aspects: parameter count, inference time, mean average precision (mAP), and floating-point operations (FLOP). The results show that YOLOv5 has the fewest parameters and the shortest inference time, with an mAP value ranking in the middle. This may be due to YOLOv5's design philosophy, which is known for being lightweight and real-time, and its design helps it better integrate into various applications. YOLOv8 has an advantage only in terms of floating-point operations, while YOLOv10 achieved improved accuracy with an increase in parameter count. YOLOv11, due to its own significant improvements, achieved a substantial increase in accuracy while reducing the parameter count.

However, the improvement proposed in this paper not only results in a slight increase in inference time and parameter count but also achieves a higher accuracy while ensuring a reduction in floating-point operations. Considering the comparative dimensions and practical application scenarios, DCB-YOLOv11 is more suitable for use in classroom student behavior recognition.

5 CONCLUSIONS AND FUTURE WORKS

The combination of scalable education and personalized training is a key goal of intelligent education. This paper proposes a student behavior recognition model based on YOLOv11 (DCB-YOLOv11), which improves classroom student behavior recognition accuracy by incorporating DCNv4 deformable convolutions in the backbone and detection head, and adding the CBAM attention mechanism in the SPPF pooling layer. Experimental results show that the model achieves an average precision improvement of 2.1% over the original YOLOv11 on a self-constructed dataset, demonstrating superior performance. The improved model can better capture target deformation and complex features, enhance feature dependencies, and support behavior recognition of multiple students. This provides more convenient

and accurate information for teaching evaluation, promoting the integration of artificial intelligence and educational big data to improve classroom teaching.

In the future, by expanding the dataset, optimizing the model structure, achieving real-time applications, and extending to other fields, an intelligent education system can be built to further enhance the technical level and application breadth. By combining behavior analysis with multi-dimensional student data and customizing personalized teaching strategies, this will promote more intelligent and efficient education, significantly improving educational quality and effectiveness.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

FUNDING

This study was supported by the:
(1) Natural Science Foundation of Xinjiang Uygur Autonomous Region (Grant No. 2022D01A226);
(2) Regional Innovation Cooperation Project of Sichuan Province (Grant No. 2023YFQ0066).

REFERENCES

[1] Liu GL, Chen ZZ, Chen R. Real-time S-T analysis method based on speaker recognition[C]// In: 2020 2nd international conference on advanced control automation and artificial intelligence(ACAAI 2020), Wuhan, Hubei, China. 2020, 147-151 .
[2] 2024 Global Smart Education Conference. Research on Educational Technology, 2024, 45(08): 2.
[3] Zaletelj J, Košir A. Predicting students' attention in the classroom from Kinect facial and body features. EURASIP J. Image Video Process, 2017, 1-12.
[4] Zheng R, Jiang F, Shen R. GestureDet. Real-time student gesture analysis with multi-dimensional attention-based detector[C]// In Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI 2020). Yokohama, Japan. 2020, 680-686.
[5] Huang Y, Liang M, Wang X, et al. Multi-person classroom behavior recognition in teaching videos based on deep spatiotemporal residual convolutional neural networks. Journal of Computer Applications, 2022, 42(3): 736-742.
[6] Lin T Y, Dollar P, Girshick S, et al. Feature Pyramid Networks for Object Detection[C]// In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, 936-944.
[7] Zhao Q, Sheng T, Wang Y, et al. M2det: A single-shot object detector based on multi-level feature pyramid network[C]// In Proceedings of the AAAI conference on artificial intelligence, 2019, 9259-9266.
[8] Wang Z, Jiang F, Shen R. An effective yawn behavior detection method in classroom[C]// In Proceedings of the 26th International Conference on Neural Information Processing (ICONIP2019). 2019, 430-441.
[9] Guo J, Lv J, Wang R, et al. Classroom behavior recognition of teachers and students driven by deep learning models. Journal of Beijing Normal University (Natural Science Edition), 2021, 57(6): 905-912.
[10] Zheng Z, Liang G, Luo H, et al. Attention assessment based on multi-view classroom behaviour recognition. IET Comput. Vis, 202.
[11] Niu W, Sun X, Yi K. Improved YOLOv5 for skeleton-based classroom behavior recognition[C]// In: Proc of the third International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI 2022), 2023, 107-112.
[12] Zhang Z, Ao D, Zhou L, et al. Laboratory Behavior Detection Method Based on Improved Yolov5 Model[C]// In: Proc. of 2021 Int. Conf. Cyber-Physical Soc. Intell, 2021, 1-6.
[13] Yang Fan. Student Classroom Behavior Detection based on Improved YOLOv7. 2023. DOI: 10.48550/arXiv.2306.03318.
[14] Yang Fan, Tao Wang, Wang Aofei. Student Classroom Behavior Detection Based on YOLOv7+ BRA and Multi-model Fusion[C]// International Conference on Image and Graphics. Cham: Springer Nature Switzerland. 2023, 41-52.
[15] Yuwen ong, et al. Efficient deformable convnets: Rethinking dynamic and sparse operator for vision applications[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024, 5652-5661.
[16] Qi Han, Zejia Fan, Qi Dai, et al. On the connection between local attention and dynamic depth-wise convolution. 2021. DOI: https://doi.org/10.48550/arXiv.2106.04263.
[17] Sanghyun W, Jongchan P. CBAM: convolutional block attention module proceedings of the European Conference on Computer Vision (ECCV). 2018, 3-19.
[18] Xizhou Zhu, Weijie Su, Lewei Lu, et al. Deformable detr: Deformable transformers for end-to-end object detection. 2020. DOI: https://doi.org/10.48550/arXiv.2010.04159.
[19] Kim J H, Kim N, Yong Woon Park, et al. Object detection and classification based on YOLO-V5 with improved maritime dataset. Journal of Marine Science and Engineering, 2022, 10(3): 377.
[20] Wang Chien-Yao, Alexey Bochkovskiy, Hong-Yuan Mark Liao, et al. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[D]// Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, 7464-7475.

[21] Wang X. Focus on Development, Emphasize Process: Exploration of Developmental Comprehensive Quality Evaluation for High School Students. Primary and Secondary School Management, 2020, (10): 21-23.
[22] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, et al. Flexible and deformable convolution for point clouds[D]// In Proceedings of the IEEE/CVF international conference on computer vision. 2019, 6411-6420.
[23] Fagad Rasheed A, Zarkoosh M. YOLOv11 Optimization for Efficient Resource Utilization. 2024. DOI: https://doi.org/10.48550/arXiv.2412.14790.