World Scientific Research Journal Volume 9 Issue 9, 2023
ISSN: 2472-3703 DOI: 10.6911/WSRJ.202309_9(9).0011

Student Online Learning Behavior Supervision Based on TSM Behavior Recognition and Screen Recognition

Huinan Xu, Ming Lu, Lijian Qiu, Wenjing Xie and Jie Xu
School of Computer and Artificial Intelligence, Wenzhou University, China

Abstract

With the increasing popularity of online classes in recent years, the regulatory demand for students' online classes is increasing. At present, the existing learning behavior monitoring applications on the market are often limited to classroom monitoring of students, and in order to effectively monitor and analyze, 3D CNN is often used. However, its computational complexity is large and it is not suitable for deployment on terminals. Although traditional 2D CNN has low computational costs, it cannot adapt to student behavior in online classes; At the same time, it is unable to accurately cooperate with multi-dimensional student behavior recognition. Therefore, this paper proposes to insert the Time Shift Module (TSM) into 2D CNN to achieve zero calculation and zero parameter time modeling, which combines the efficiency of 2D convolution with the high performance of 3D convolution. ResNet50, a backbone network, is selected and applied to the monitoring of students' learning behavior. This article designs a screen capture recognition algorithm to assist in monitoring the learning behavior of the aforementioned students, and achieves a comprehensive judgment accuracy of 98.61% for the students' learning situation. And a simple desktop application was created to verify the feasibility of the solution, and the running time on the CPU was measured to be about 106ms.

Keywords

Learning behavior; Time shift module; Online learning; Deep learning; Video analysis.

1. INTRODUCTION

In recent years, due to the pandemic, online teaching has become increasingly important in the learning content of middle school students. However, for online learning at home, parents often cannot effectively supervise their children; Teachers are also unable to timely understand students' learning status on the other end. Therefore, a mobile student behavior supervision method is needed.

At present, most student behavior supervision often involves capturing students in the classroom environment through high mounted cameras, including videos and images. Then, the captured videos or images are annotated and classified, and fed into the neural network for training. In China, Hu Liping[1] divided students' learning behavior into six categories by collecting a dataset of pictures: listening, writing, lying on the table, raising hands, standing, and looking left and right. Select ResNet50 as the basic classification network to classify student behavior, and use Python's Tkinter library to quickly construct a visual display system for behavior recognition results. Zhu Chao [2] uses deep learning techniques to detect and recognize students' behavioral states from the perspective of lowering their heads and raising their heads by extracting student target images. The above image based deep learning strategies, although having high accuracy and speed, are not fully applicable to online student learning behavior recognition. By analyzing the learning behavior of students in online learning, it was

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202309_9(9).0011

found that the frequency of students looking up and down at the computer was significantly higher than that of offline learning. If students were to randomly capture their images directly through the front camera of the computer, there would be multiple instances of unclear capture of their learning status. In summary, simply capturing and recognizing individual student images is not suitable for monitoring students' online learning behavior. Tao Yaping[3]constructed a dataset of student classroom behavior videos and analyzed and divided teaching videos from real classroom teaching into seven typical classroom behaviors: listening, writing, flipping, standing, sleeping, playing with mobile phones, and looking around. Based on the dual attention mechanism and the integration of spatiotemporal features, an improved behavior recognition algorithm was used to recognize student behavior and achieved high accuracy; Xie Wei et al. [4] transferred the YOWO model in behavior recognition to classroom students' learning behavior recognition scenarios; Finally, through experimental comparison with the VGG16 network model based on image datasets and the YOLO-V2 network model based on video datasets, it was verified that the YOWO model has a high accuracy in identifying students' learning behavior in classroom videos. However, the above student learning behavior recognition is conducted in the classroom, with multiple students and multiple objectives. Online learning is usually a student, and the shooting angle is also significantly different from the classroom. Following the above models and related methods, there are good training results on datasets that cannot be learned online. At the same time, traditional video learning deployment on the mobile end is too large, the recognition time is too long, and the computational power required is relatively large. Traditional video deep learning strategies, performance is not suitable for online learning behavior supervision. In addition, although students are focused on staring at computer screens, it is also possible to watch movies, play games, etc. through the computer, so it is necessary to monitor the screen and comprehensively recognize students' learning behavior in order to accurately monitor. Lin et al. proposed a TSM (Temporary Shift Module) module that can be used in both offline and online scenarios. By moving feature map channels along the time dimension to extract spatiotemporal features, temporal information can be obtained while maintaining the original parameter quantity and computational cost of 2DCNN.

In response to the aforementioned problems, this article proposes a method that combines the TSM[5]module with ResNet50 and conducts training; At the same time, by capturing desktop videos and preprocessing them, analyzing their color change rate and motion features, and adding weights to the analysis results as a threshold to determine whether they are related to learning. By establishing a student online learning behavior dataset, a single TSM-ResNet50 model achieved an accuracy of 95.88% in monitoring student online learning behavior, and the real-time online behavior monitoring accuracy of the two modules was measured to be 98.61%. In addition, a simple client has been developed and deployed on a computer to verify its feasibility. Testing has shown that the minimum running time on the CPU is about 106ms.

2. NETWORK STRUCTURE

2.1. Backbone

When performing deep learning classification on videos, the accuracy and performance of traditional neural networks may decrease with the increase of the number of deep learning network layers, and there may be phenomena of gradient vanishing and exploding. ResNet introduces residual modules to implement deep network modeling to alleviate this phenomenon. Its definition is:

$$y=F（x，\{Wi\}）+x \quad (1)$$

Where y represents the input of the residual module; X represents the input of the residual

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202309_9(9).0011

module, and F (x, {Wi}) represents the residual mapping. The residual block is an important component of ResNet, and its structure is shown in Figure 1.

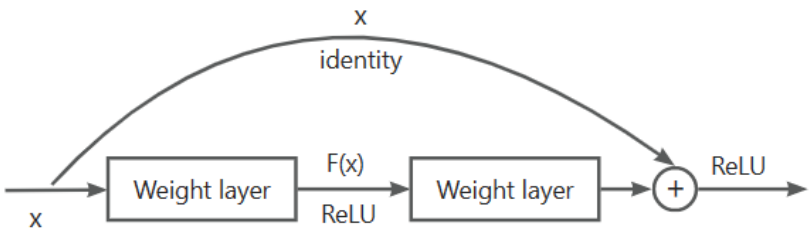

Figure 1. Residual module structure

Considering the need to monitor students' learning behavior in real-time, the computational complexity of the model should not be too large, but the algorithm performance should not be too poor. Compared to ResNet18 and ResNet101, this article uses the ResNet50 network, which is more suitable for datasets and tasks. Its depth and complexity are at a moderate level in many models, making it neither as fast as shallow models to reach the performance limit nor as complex as overly deep models. The specific comparison results are shown in 2.4. The relevant network structure of ResNet50 is shown in Table 1.

Table 1. ResNet50 Network Architecture

Layer name	Output size	18-layer	34-layer	50-layer	101-layer	152-layer
Conv1	112×112	7×7,64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix}3\times3,64\\3\times3,64\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,64\\3\times3,64\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,64\\3\times3,64\\1\times1,256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,64\\3\times3,64\\1\times1,256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,64\\3\times3,64\\1\times1,256\end{bmatrix}\times3$
Conv3_x	28×28	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,128\\3\times3,128\\1\times1,512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1,128\\3\times3,128\\1\times1,512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1,128\\3\times3,128\\1\times1,512\end{bmatrix}\times8$
Conv4_x	14×14	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,256\\3\times3,256\\1\times1,1024\end{bmatrix}\times6$	$\begin{bmatrix}1\times1,256\\3\times3,256\\1\times1,1024\end{bmatrix}\times23$	$\begin{bmatrix}1\times1,256\\3\times3,256\\1\times1,1024\end{bmatrix}\times36$
Conv5_x	7×7	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,512\\3\times3,512\\1\times1,2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,512\\3\times3,512\\1\times1,2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1,512\\3\times3,512\\1\times1,2048\end{bmatrix}\times3$
	1×1	Average pool, 1000-d fc, softmax				
FLOPs		1.8×109	3.6×109	3.8×109	7.6×109	11.3×109

2.2. TSM(Temporal Shift Module)

Traditional 3D convolutional neural networks use 3D convolution in both spatial and temporal dimensions, requiring a large amount of computational resources and high deployment costs when processing video data. In contrast, TSM does not perform convolution operations on the temporal dimension, but instead achieves the exchange of temporal information by moving information on the feature map. At the same time, it can be inserted into

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202309_9(9).0011

2D CNN to achieve zero computation and zero parameter time modeling, which combines the efficiency of 2D convolution with the high performance of 3D convolution.

Taking a one-dimensional convolution with a kernel size of 3 as an example, the weight of the convolution is W=(w1, w2, w3). Input X is a 1D infinitely long vector. Integrate the volume into two steps, with the first step being displacement, which does not require time cost and is specifically defined as:

$$Y = w1X-1 + w2X0 + w3X+1 \quad (2)$$

The second step is multiplication accumulation. TSM combines the multiplication accumulation part into 2D convolution, so there is no additional overhead compared to traditional 2D CNN. The specific TSM displacement process[6]schematic diagram is shown in Figure 3.

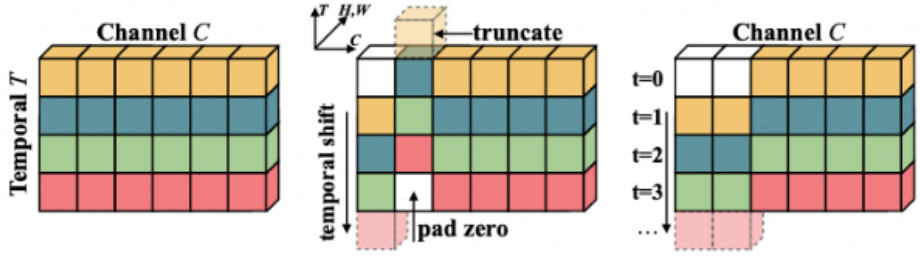

Figure 2. Schematic diagram of TSM displacement process

The movement of TSM will result in a decrease in the spatial modeling ability of the model. To improve this issue, TSM is placed in the residual branch of the residual module, which can solve the problem of degraded spatial feature learning, as the original activation information can still be accessed through identity mapping after time transfer. The specific form is shown in Figure 3.

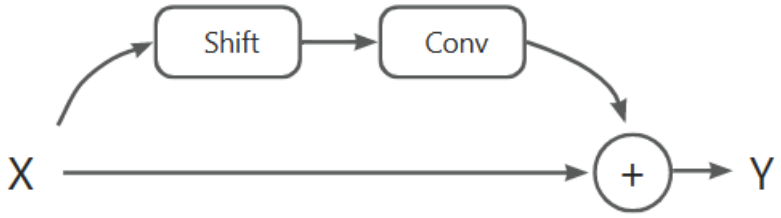

Figure 3. TSM put into residual branch

3. EXPERIMENTATION

3.1. Experimental environment

The hardware configuration for the experiment is CPU 5-core E5-2680 v4, GPU RTX 3060, with a total of 12.6GB of graphics memory. The software environment is the Ubuntu operating system, with CDUA version 10.1 and cudnn version 7.6.4. The deep learning framework uses PaddlePaddle, and Python version 3.7.

3.2. Dataset establishment and data augmentation

Currently, there is no relevant dataset for individual students' online learning on the market. Therefore, this article has established its own dataset for individual students' online learning.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202309_9(9).0011

Part of the dataset sources include project members collecting through the internet, while the other part is recording by searching for volunteers. Due to the varying resolutions of volunteers' laptop cameras, the duration of recorded videos, and the frame rate and resolution of videos collected online, the videos were pre processed after obtaining the original materials. Trim the video using Python's moviepy library to make it a video with a length of 5 seconds, a frame width of 320, and a frame height of 240. After relevant processing, a total of 5000 video segments were obtained. Divide 5000 items into four categories: studying with your head down, looking up at the screen, standing, and eating snacks.

The screen recognition dataset comes from online game videos and movie video clips; Online course videos from some online learning platforms. Process the data recorded on the screen into a 5-second video with a frame width of 320 and a frame height of 240. Divide videos into two categories: learning related and non learning related, with a total of 40000 segments.

In order to expand the training dataset and improve the generalization and robustness of the model, multi-scale pruning and random flipping are performed on the data in the training dataset.

The test set consists of a 5-second screen recording video and a 5-second student behavior video, with a frame width of 320 and a frame height of 240, totaling 800 videos and 1600 videos.

3.3. TSM parameters

The number of channel displacements in TSM has a crucial impact on the recognition accuracy of the TSM-ResNet50 model. This article tested the impact of multiple channel displacements on recognition accuracy, with displacements of 1/8,1/4,1/2. The specific test results are shown in Figure 4. Due to the fact that displacement requires almost no time cost, this article adopts the 1/8 displacement with the highest recognition accuracy.

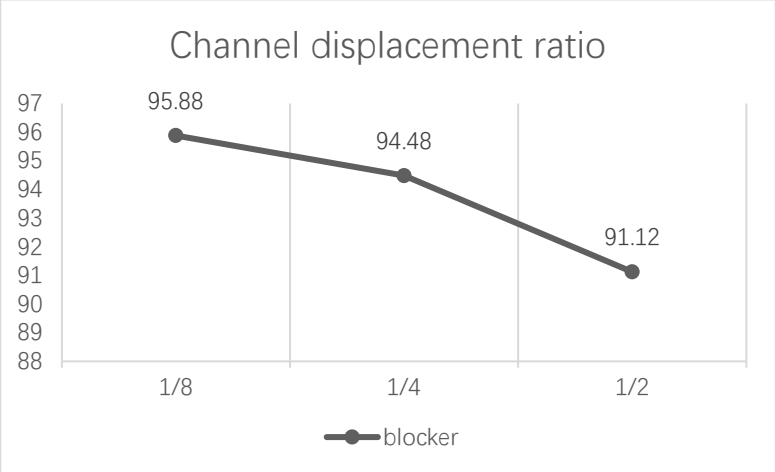

Figure 4. Channel displacement ratio and recognition accuracy

3.4. Model training and analysis of experimental results

TSM-ResNet50 only uses a dataset of student learning behavior during training and does not include screen recognition.

In this experiment, a total of four different layers of backbone networks were used: ResNet18, ResNet34, ResNet50, ResNet101, and a small batch random gradient descent training method was selected. After comparison, the final batch size was determined to be 32.

The initial learning rate is 0.001. As the number of iterations continues to increase, the learning rate will decay at a rate of 10 times to 0.0001 and 0.00001 at the 11th and 21st epochs.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202309_9(9).0011

The evaluation standard for video behavior recognition generally adopts top-k accuracy, and this article uses top-1 accuracy as the evaluation indicator.

Before model training, it is necessary to preprocess the data to ensure that the model can read the data normally. These operation steps include: decoding the video into frames, frame sampling (segmenting the video, randomly selecting a starting position from each segment, and collecting 8 consecutive frames from the selected starting position), image scaling, multi-scale cropping, random flipping, data format conversion (converting the data format from PIL. Image to Numpy), normalization Center cropping (similar to random cropping, with the difference being the method of selecting the starting point for cropping). Among them, for the training mode, perform multi-scale cropping and random flipping operations, and for the test mode, perform image scaling and center cropping operations. The workflow of the model is shown in Figure 5.

Based on the comparison of ResNet networks with different layers, if there are too few layers, the accuracy can be significantly reduced, while if there are too many layers, the latency can be significantly increased. The experimental results are shown in Table 2. Therefore, this article ultimately chooses ResNet50 as the backbone network.

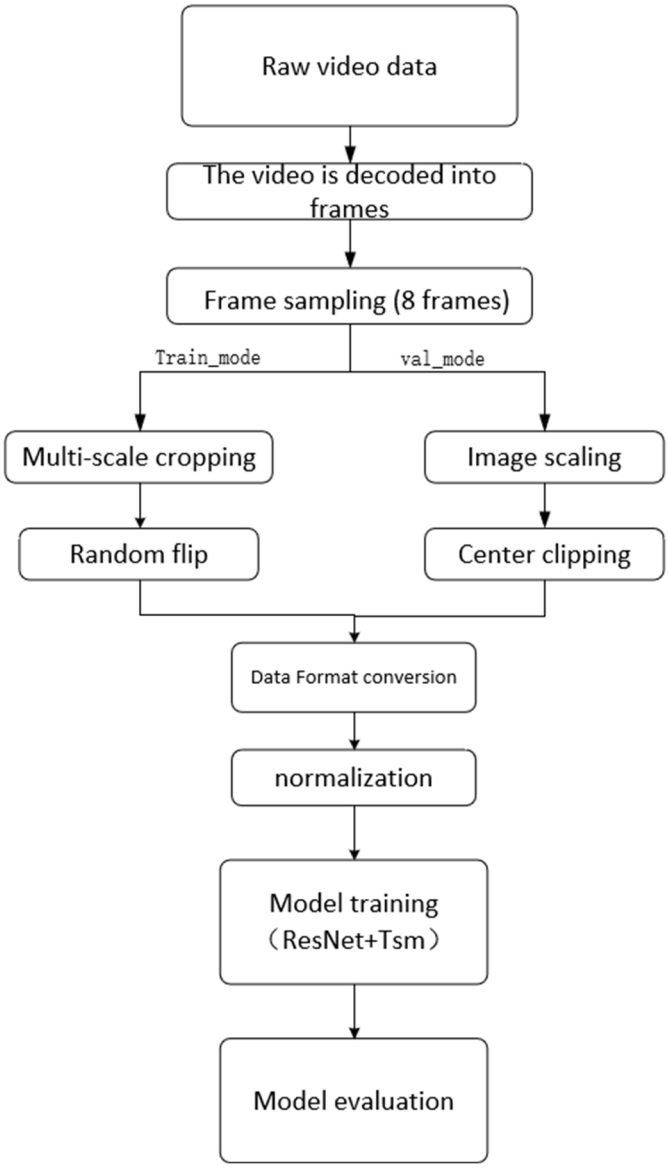

Figure 5. Model workflow

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202309_9(9).0011

Table 2. Experimental result

backbone	Top1 accuracy	Loss	Delay/ms
ResNet18	87.69%	0.6023	53.20
ResNet34	92.56%	0.4844	69.03
ResNet50	95.88%	0.4798	84.38
ResNet101	96.74%	0.3596	125.44

3.5. Screen recognition

Although the accuracy of student learning behavior recognition is already high, there are certain differences between online learning and traditional classroom learning. For example, although students do not engage in activities unrelated to their studies, such as walking or eating snacks, under behavioral supervision. But looking up carefully at the screen still cannot determine whether you are studying, such as focusing on watching movies or playing games. Recognition of front facing cameras alone is not enough, it must be combined with screen recognition to be effective and reliable.

If a deep learning strategy for screen content classification is added to the existing TSM-ResNet50 model, it will undoubtedly increase the time cost and go against the original intention of efficient deployment and prediction. After repeated research on the features of videos, it was found that there is a significant difference in the features of learning related and learning unrelated content, and the recognition of the two can be achieved with high accuracy without the use of deep learning. To solve this problem, this article proposes a method: extracting color change rate and motion features from existing screen classification datasets, obtaining the average and variance, and adding weights to set thresholds, plus bias coefficients. If it is above or equal to the threshold, it is considered that the screen is playing content unrelated to learning, and vice versa, it further determines the student's behavior.

This article classifies 40000 prepared videos and analyzes their color change rate. By averaging each video, it is found that the color change rate of games is generally higher than 8, while the color change rate related to learning is generally lower than 5. Extracting motion features from the dataset, as the motion features of objects or characters in game videos are more obvious, the opposite is true for learning related videos. Therefore, this article mainly extracts the intensity of optical flow and calculates the direction of optical flow, where the numerical value of the direction of optical flow ranges from 0 to 2π. And calculate the variance for each video. Through comprehensive comparison, it was found that the changes in sports features such as games and movies were greater, with a greater variance, generally higher than 3, while learning related videos were the opposite, generally lower than 2. The weight values for color change rate and motion features are 70% and 30%, respectively. As screen recognition is mainly used to assist in behavior recognition and judgment in some special situations, the threshold should be high to prevent misjudgment. Therefore, take b as 0.8, and the specific calculation formula is as follows. The threshold is calculated to be 8.5. If the value exceeds 8.5, it is determined that the desktop is playing something unrelated to learning.

$$T = C(x)W_1 + M(x)W_2 + b \tag{3}$$

3.6. Desktop applications

This article applies Python's tkenter library to create relevant interfaces and calls trained models for online testing. The interface includes three modules: desktop content displayed on the left, student behavior recognized and displayed through camera devices on the right, and behavior discrimination results and time delay displayed in the text below. This interface is only

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202309_9(9).0011

used to validate the model mentioned earlier. The measured latency is 121ms, which verifies the feasibility of deployment on the terminal. The identification test is shown in Figure 6. During actual deployment, the front facing camera of the machine can be called, or the device can be remotely connected through the network for parents to view in real-time, or corresponding functions can be added as needed.

4. CONCLUSION

This article proposes a method for student online learning supervision that combines the TSM module with the ResNet50 backbone network and applies it to student learning behavior recognition. Simultaneously using screen capture and recognition as auxiliary tools to address the unique special situations of online learning, relevant system interfaces were developed to verify its feasibility. The experimental results show that the proposed model achieves high accuracy and low latency for high deployability.

ACKNOWLEDGMENTS

This paper was supported by National Undergraduate Innovation and Entrepreneurship Training Program of Wenzhou University in 2022 (202210351060); Innovation and entrepreneurship project for college students of Wenzhou University in 2022: ”Research on Student Learning Behavior Supervision Based on Deep Learning” (JWXC2022175).

REFERENCES

[1] Hu Liping A Study on the Model of Classroom Behavior Recognition for Students Based on Deep Learning [D]. Hangzhou Normal University, 2021. DOI: 10.27076/d.cnki. ghzsc.2021.000555

[2] Zhu Chao Recognition and Application of Classroom Bowing and Heading Up Behavior Based on Deep Learning [D]. Central China Normal University, 2019. DOI: 10.27159/d. cnki. ghzsu. 2019. 000266

[3] Tao Yaping Classroom learning behavior recognition based on spatiotemporal characteristics [D]. Central China Normal University, 2022. DOI: 10.27159/d. cnki. ghzsu. 2022-000615

[4] Xie Wei, Tao Yaping, Gao Jie, et al. Real time recognition of classroom learning behavior based on YOWO [J]. Modern Education Technology, 2022,32 (06): 107-114

[5] TSM: Temporal Shift Module for Efficient Video Understanding.[J]. IEEE transactions on pattern analysis and machine intelligence,2020,PP.

[6] He Zhizhou, Miao Yubin, Zhang Yonghang, et al. Research on online recognition of hand washing actions based on TSM-MobileNetV3 [J]. Mechatronics, 2022-28 (Z2): 3-10. DOI: 10.16413/j.cnki. issn.1007-080x.2022z2.001