Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 5303503, 10 pages
https://doi.org/10.1155/2022/5303503

Research Article

Lecture Video Automatic Summarization System Based on DBNet and Kalman Filtering

Fan Sun [1,2,3] **and Xuedong Tian** [1,2,3]

[1]*School of Cyber Security and Computer, Hebei University, Baoding 071002, China*
[2]*Hebei Machine Vision Engineering Research Center, Hebei University, Baoding 071002, China*
[3]*Institute of Intelligent Image and Document Information Processing, Hebei University, Baoding 071002, China*

Correspondence should be addressed to Xuedong Tian; xuedong_tian@126.com

Received 3 June 2022; Revised 19 July 2022; Accepted 29 July 2022; Published 31 August 2022

Academic Editor: Zhihan Lv

Video summarization for educational scenarios aims to extract and locate the most meaningful frames from the original video based on the main contents of the lecture video. Aiming at the defect of existing computer vision-based lecture video summarization methods that tend to target specific scenes, a summarization method based on content detection and tracking is proposed. Firstly, DBNet is introduced to detect the contents such as text and mathematical formulas in the static frames of these videos, which is combined with the convolutional block attention module (CBAM) to improve the detection precision. Then, frame-by-frame data association of content instances is performed using Kalman filtering, the Hungarian algorithm, and appearance feature vectors to build a tracker. Finally, video segmentation and key frame location extraction are performed according to the content instance lifelines and content deletion events constructed by the tracker, and the extracted key frame groups are used as the final video summary result. Experimenting on a variety of scenarios of lecture video, the average precision of content detection is 89.1%; the average recall of summary results is 92.1%.

1. Introduction

The rapid development of computer technology and online education means video has become an important resource for students and educators. The impact of the spread of COVID-19 on traditional educational methods also makes online video education play an increasingly important role. In a large number of lecture videos, contents such as texts and mathematical expressions can often summarize and locate the videos. Automatically extracting and summarizing these contents can effectively utilize educational video resources and enable users to quickly browse the contents. The online education system can also conduct effective content management of video assets through the technology of lecture video summarization to achieve functions such as indexing, browsing, retrieval, and promotion. Based on these needs, the research on lecture video content summarization technology is extremely valuable.

For general video summarization, there are many methods that use a set of automatically extracted key frames to represent the main content of the video [1, 2]. These methods seek to find important scenes, objects, colors, and moving objects in videos and usually follow three steps, namely, video feature extraction, frame image clustering [3, 4] or classification, and key frame selection. However, these methods do not scale well to lecture videos. A semantically meaningful change, such as text popping up in a slideshow, usually results in a rather subtle appearance change in the video and is thus ignored by these methods. On the other hand, the lecturer's position movement can cause significant appearance changes, triggering extraneous key frames.

In these videos, lecturers usually use projection to demonstrate the learning content or use blackboard, whiteboard, paper, or electronic device screen for handwritten interpretation. Content extraction faces challenges such as complex backgrounds and occlusion. Also,

mathematical formulas have complex two-dimensional structures, and courses with a lot of math content are more inclined toward handwritten demonstrations. Therefore, the methods of automatic speech recognition technology are not fully applicable.

Traditional lecture video summarizations generally design algorithms based on teaching scene features. For academic videos based on slide presentations, Li et al. [5] proposed a fully automated system to extract the semantic structure of academic slide presentation videos, the system automatically locates and tracks the projection screen, tracks the sparse optical flow feature points in the screen region, detects the slide progression by analyzing the feature point trajectories, constructs a frame index with a large number of feature appearances or disappearances, and extracts for each slide a high quality, nonoccluded, geometrically compensated images to generate a representative set of image lists that reconstruct the main presentation structure of the slide, and experimental results show that for this specific type of video, the system is able to extract a more accurate representation structure than general video summarization methods. Davila and Zanibbi [6] first locate the whiteboard region, use the lag image between Otsu's binarization and random forest binarizer to generate binary images of whiteboard handwriting, generate spatio-temporal indices for handwriting, and detect and eliminate content. Conflicts between regions are time-segmented to extract key frames, and tests on the AccessMath dataset show that the summary method has a good compression ratio. Rahman et al. [7] proposed a new visual summarization method for lecture videos by dividing the video into multiple segments based on the inter-frame similarity of the content and defining the most representative images by estimating the importance of each image in the segment, calculating the distance matrix between images, and using a graph-based algorithm; the proposed algorithm is significantly better than random selection and cluster-based selection, and only slightly lower than manual selection.

With the development of computer vision and deep learning, much research is based on neural networks. Dutta et al. [8] investigated the effectiveness of state-of-the-art scene text detection networks for text detection in lecture video scenes and built LectureVideoDB, a static frame dataset of English lecture videos for this purpose; experimental results show that existing methods perform poorly on this dataset and need to be improved for application in educational scenes; in this work, the EAST scene text detection model [9] was used as a baseline to develop a system for detecting and recognizing instructional video text, but mathematical expressions and sketches as important elements were not annotated and evaluated. Since the lecturers will perform various actions with semantic information, such as writing and erasing, during the teaching process, Xu et al. [10] proposed a method based on speaker action classification, using the OpenPose pose estimator [11] to extract body and hand skeletal data to calculate action features and then using random forests and motion features to classify speaker actions, segmented the video based on handwritten content erasing actions to extract key frames from lecture videos of handwritten whiteboard content as video summary, and the summary results with good compression. Davila et al. [12] proposed an FCN-LectureNet model based on a fully convolutional neural network (FCN) to extract English handwritten content from videos as binary images, further generate a time-space index of handwritten content, and create key frame-based handwritten content summaries based on the time periods that change when a large amount of content is deleted, and validation results showed that this method outperforms some existing handwritten lecture video summarization methods.

To sum up the above, most of the lecture video summarization methods based on visual content extraction are aimed at some specific scenes, such as slide teaching scenes and whiteboard handwriting teaching, and are mainly in English, which has a certain impact on the robustness and generalization performance of the system. To address the above difficulties, this paper improves the deep learning-based text detection algorithm and expands the Chinese teaching video dataset to detect text and mathematical formulas in a variety of teaching scenarios. Use the Kalman filtering and Hungarian algorithm to track content instances, construct content instance lifelines to segment lecture video based on the tracking result, and complete the positioning and summary of lecture video key frames. The main contributions of this paper are as follows:

(1) Combining DBNet [13], a scene text detection network with differentiable binarization method, with convolutional block attention module (CBAM) [14], which has spatial and channel attention mechanisms, adapts DBNet to the detection of text, mathematical formulas, and sketches in static frames of instructional videos to improve detection precision.

(2) A multi-target tracking method based on Kalman filtering and the Hungarian algorithm is introduced for content instance tracking, and adding content instance appearance vector matching before geometric position matching improves the tracking method and reduces the false tracking caused by simple geometric position matching.

(3) Lecture videos of advanced education lectures taught in Chinese in various scenarios are collected to build the dataset. On the video still frames, content such as text is annotated for content detection training; key frames are manually selected for comparison with the automatically extracted key frames.

The rest of the paper is structured as follows: the second part elaborates the lecture video summarization method of this paper; the third part analyzes and discusses the experimental results; the fourth part summarizes the paper.

2. Materials and Methods

The overall flowchart of the proposed lecture video summarization method is shown in Figure 1.

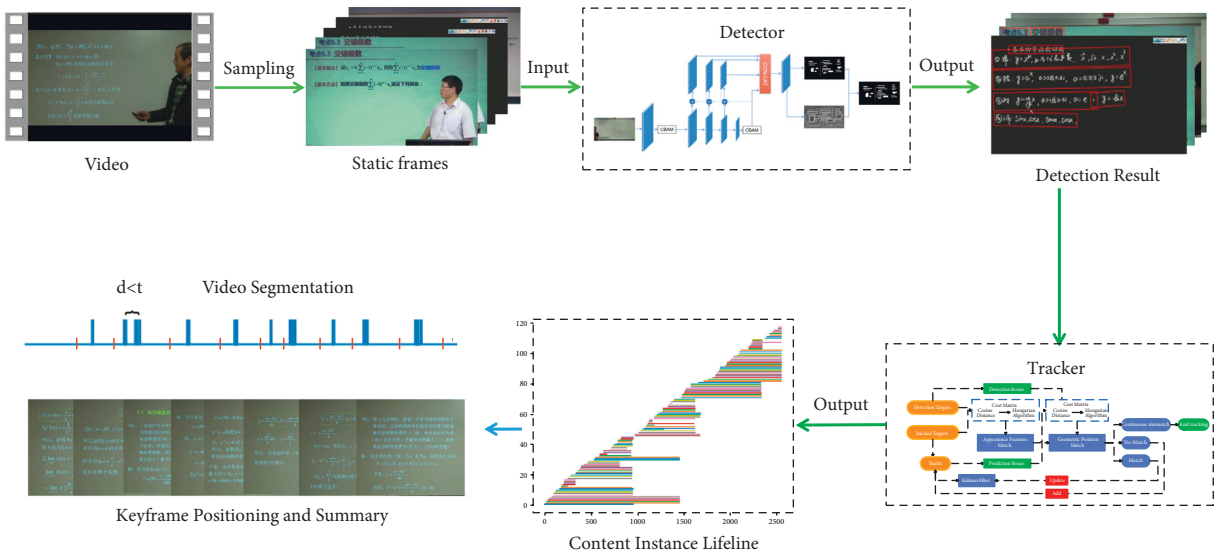


Figure 1: The flow of the method proposed in this paper.

2.1. CBAM-DBNet Content Detector. The real-time text detection network DBNet with a differentiable binarization method is used as a lecture video content detector to detect the text and mathematical formulas in the lecture video. The DBNet backbone network adopts ResNet [15] and uses deformable convolution [16] in the conv3-conv5 layers for feature extraction. Deformable convolution can adaptively obtain the morphological features and scale information of the target. Deformable convolution can adaptively obtain morphological features and scale information of the target, which facilitates the detection of contents with extreme aspect ratios in still frames of lecture videos. The feature pyramid networks (FPNs) [17] are used to upsample the conv2-conv5 layers and perform feature fusion to deal with the multi-scale variation in detection; in the output part of the network, the approximate binarization map is calculated using the probability map P and the adaptive threshold map T predicted during the training process, and the detection bounding box is inferred from the approximate binarization map.

Due to the existence of complex background, image noise, and occlusion in teaching scenes, in order to increase the differentiation between content and noncontent regions, this paper adds the convolutional block attention module (CBAM) after the cov1 and cov5 layers of the backbone network of DBNet to construct the CBAM-DBNet content detector for spatial and channel attention to make the network pay more attention to target objects such as text and mathematical formulas in feature extraction of static frame images. CBAM is added to the first and last convolutional layers of ResNet in order to be able to use pretraining parameters without changing the network structure. The structure of the CBAM-DBNet detection network is shown in Figure 2.

The differentiable binarization method of DBNet and the convolutional block attention module (CABM) are introduced as follows.

2.1.1. Differentiable Binarization (DB). In the DBNet algorithm, the binarization operation is inserted into the segmentation network for joint optimization in order to adaptively predict the threshold value at each position of the image in order to better distinguish the foreground and background regions. However, the traditional standard binarization function is not differentiable; a differentiable approximate binarization function, called differentiable binarization, is given in DBNet so that the binarization operation can be trained together with the segmentation network. The standard binarization and differentiable binarization are shown in

$$B_{i,j} = \begin{cases} 1, & P_{i,j} >= t, \\ 0, & \text{otherwise}, \end{cases} \tag{1}$$

$$\hat{B}_{i,j} = \frac{1}{1 + e^{-k\left(P_{i,j} - T_{i,j}\right)}}. \tag{2}$$

2.1.2. Convolutional Block Attention Module. The convolutional block attention module (CBAM) is a lightweight, general-purpose feedforward convolutional neural network attention module that contains the spatial attention module (SAM) and the channel attention module (CAM). The structure of CBAM is shown in Figure 3.

Given the feature map $F \in \mathbb{R}^{C \times H \times W}$ as input, CBAM inferred the 1D channel attention map $P_{ca} \in \mathbb{R}^{C \times 1 \times 1}$ and 2D spatial attention map $P_{sa} \in \mathbb{R}^{1 \times H \times W}$ in turn, and the overall attention process is shown in

$$F' = P_{ca}(F) \otimes F, \tag{3}$$

$$F'' = P_{sa}(F') \otimes F'. \tag{4}$$

The channel attention module, in order to calculate the importance of different feature channels more efficiently, compresses the input feature map F through the average pooling layer and the maximum pooling layer, respectively, and turns the feature map of size $C \times H \times W$ into two feature

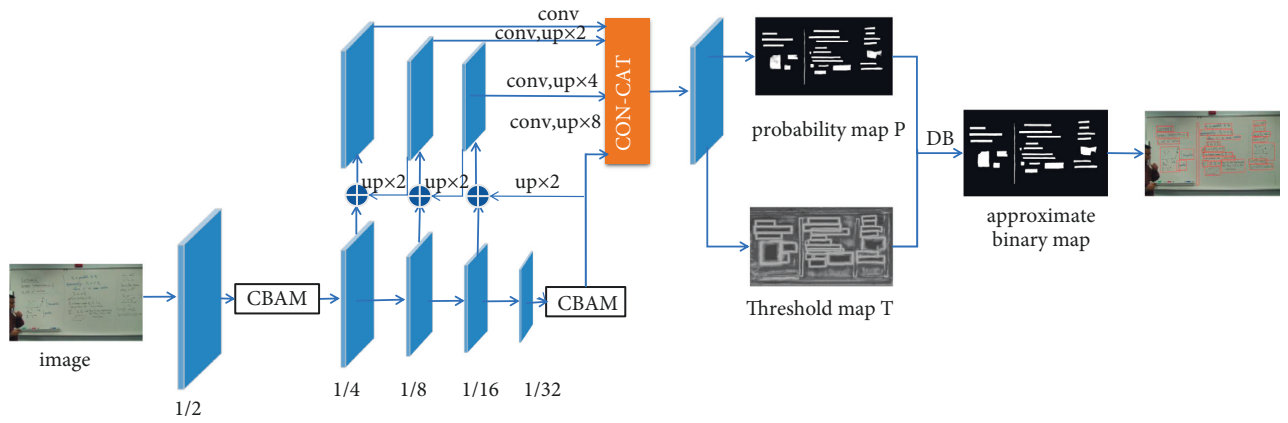


Figure 2: The structure of the improved content detection network.

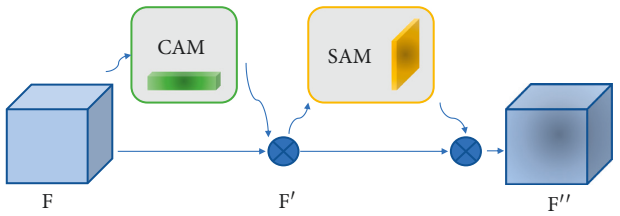


Figure 3: Convolutional block attention module structure.

maps of size $C \times 1 \times 1$. The compressed two feature maps are convolved by a shared multilayer perceptron (MLP) operation, and the output results are summed at the element level and activated by the sigmoid activation function to obtain the feature map $P_{ca}(F) \in \mathbb{R}^{C\times 1\times 1}$ with channel attention weights. $P_{ca}(F)$ and the original feature map F are multiplied by channel to obtain the new feature map F' with channel attention weighting. To calculate spatial attention, the feature map F is first passed through maximum pooling operation and average pooling operation, respectively, to form two feature vectors of size $1 \times H \times W$, and the two features are connected together to form a feature map of size $2 \times H \times W$. Then, through a convolutional layer, the feature map dimension changes from $2 \times H \times W$ to $1 \times H \times W$. The $1 \times H \times W$ feature map characterizes the importance of each point on the feature map and is activated using the sigmoid function to generate a feature map $P_{sa} \in \mathbb{R}^{1\times H\times W}$ with spatial attention weights. Then, P_{sa} is multiplied with F' to obtain the feature map F'' with channel attention and spatial attention weighting as the output of the CBAM.

2.2. Tracker for Content Instances. Introducing Kalman filtering and the Hungarian algorithm to deal with position prediction and inter-frame data association in content instance tracking, respectively, Kalman filtering and the Hungarian algorithm have played a significant role in the field of multi-target tracking [18–20]. In this paper, the appearance feature matching module is added to the Kalman filtering and Hungarian algorithm-based multi-target tracking algorithm [18] to integrate appearance features and geometric location features for content instance tracking and reduce the false tracking caused by simple geometric location matching. The tracking process is shown in Figure 4.

The specific steps of content instance tracking are described as follows:

(1) The initial frame detection result is used as the tracked target of the tracker, and the Kalman filter is initialized. Kalman filtering propagates the tracked content instance target state to the subsequent frames, correlates the detection result of the current frame with the tracked target, and manages the tracked target. The state of the target is modeled as shown in

$$M = [h, v, a, r, \dot{h}, \dot{v}, \dot{a}, \dot{r}]^T, \tag{5}$$

h and v represent the pixel position of the center of the target bounding box; a and r represent the pixel size and aspect ratio of the target bounding box, respectively; $(\dot{h}, \dot{v}, \dot{a}, \dot{r})$ corresponds to the motion speed of the (h, v, a, r) components between the front and rear frames.

(2) After embedding the content instance representation information extracted by ResNet18 into a vector δ, the cosine distance is used to calculate the similarity between the representation vector stored in the track and the detection result representation vector of the current frame. The cosine distance measurement formula based on appearance features is shown in

$$d_{\text{cha}}(i, j) = 1 - \delta_j^T \delta_i, \tag{6}$$

i and j represent the i-th trajectory stored in the tracker and the j-th result detected by the detector, respectively. The cost matrix of the Hungarian algorithm is constructed with $d_{\text{cha}}(i, j)$ for appearance feature matching of content instances. The Hungarian algorithm is a data association algorithm that seeks the maximum match. It obtains the maximum matching pair within the matching threshold according to the cost matrix and the principle of minimum cost. The smaller the $d_{\text{cha}}(i, j)$ is, the more

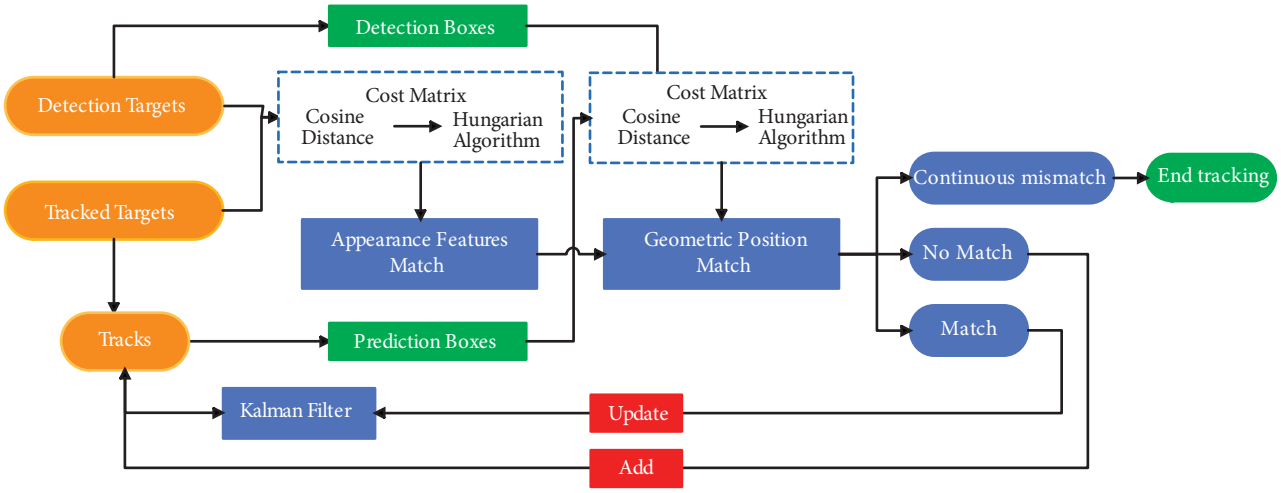


Figure 4: The flow of content instance tracking.

similar the two appear, and the more likely they are the same tracking target.

(3) The Kalman filtering uses the (h, v, a, r) of the tracking target as a variable to predict the target state of the current frame and uses the IOU (intersection and union ratio) between the predicted $bbox_{pre}$ set and the detector's detection result $bbox_{det}$ set to calculate the geometric similarity. Based on IOU, the geometrical position distance measurement formula is shown in

$$d_{\text{geo}}(i, j) = 1 - \frac{\text{area}\left(b\text{box}_i \cap b\text{box}_j\right)}{\text{area}\left(b\text{box}_i \cup b\text{box}_j\right)}, \tag{7}$$

$d_{\text{geo}}(i, j)$ is used to construct the cost matrix of the Hungarian algorithm for geometric position matching of content instances.

(4) If the geometric position matching result is consistent with the appearance feature matching result, the matching is successful, and the status of the trajectory is updated. If the tracking target fails to match continuously on subsequent frames for more than Fmax frames, the tracking of the track is ended.

2.3. Video Segmentation and Summaries. According to the content instance tracking results, the trajectories of all tracked targets on the time axis in a complete teaching video are obtained, including interference such as character occlusion, as shown in Figures 5(a) and 5(b). The lifeline of the content instance on the video timeline is constructed based on the start and end times of the target trajectory, as shown in Figures 5(c) and 5(d).

In order to extract a static summary of an instructional video, that is, a set of key frames that best summarize the video content, it is first necessary to divide the video into time segments with semantic information. The semantic time segments of instructional videos are usually updated to a set of handwritten or projected instructional content. To start, end with the group of instructional content disappearing from the video. In this paper, inspired by Xu et al. based on identifying speaker action erasure events and the FCN-LectureNet method based on main content deletion events for video time segmentation, the end of the content instance lifeline is used as the signal, and the cumulative deletion events on the video timeline are the basis for video segmentation. On the video timeline, the visualization of the normalized content added, deleted, and total area size is shown in Figure 6.

After the video is divided into several time sub-segments, the static frames containing all the track objects of the current segment are extracted as key frames in each segment interval, as shown in Figure 7. A set of key frames extracted from a complete instructional video is used as the summary of the video.

3. Results and Discussion

3.1. Introduction to the Dataset. The dataset contains 5 Chinese online advanced mathematics lecture videos collected on the Internet, including a variety of scenes and content forms (projected and handwritten), some static frames of the 4 videos are marked with content instances such as text and mathematical formulas, and manual key frame selection is performed for each video. To complement the variety of lecture video scenarios, three English whiteboard handwritten lecture videos from the publicly available dataset AccessMath [21] were selected. The information for each video is shown in Table 1.

In the training phase of the detection network, 4648 images were randomly selected as the training set and 1510 images were used as the test set; during preprocessing, data enhancement was performed by randomly cropping the image size to 640×640 and randomly rotating $(-10^{\circ}, 10^{\circ})$.

3.2. Content Detection Evaluation Index and Experimental Results. Using recall, precision, and F1-score as evaluation metrics for content detection network, the evaluation method uses the scene text detection evaluation method DetEval [22], which considers three types of rectangular box

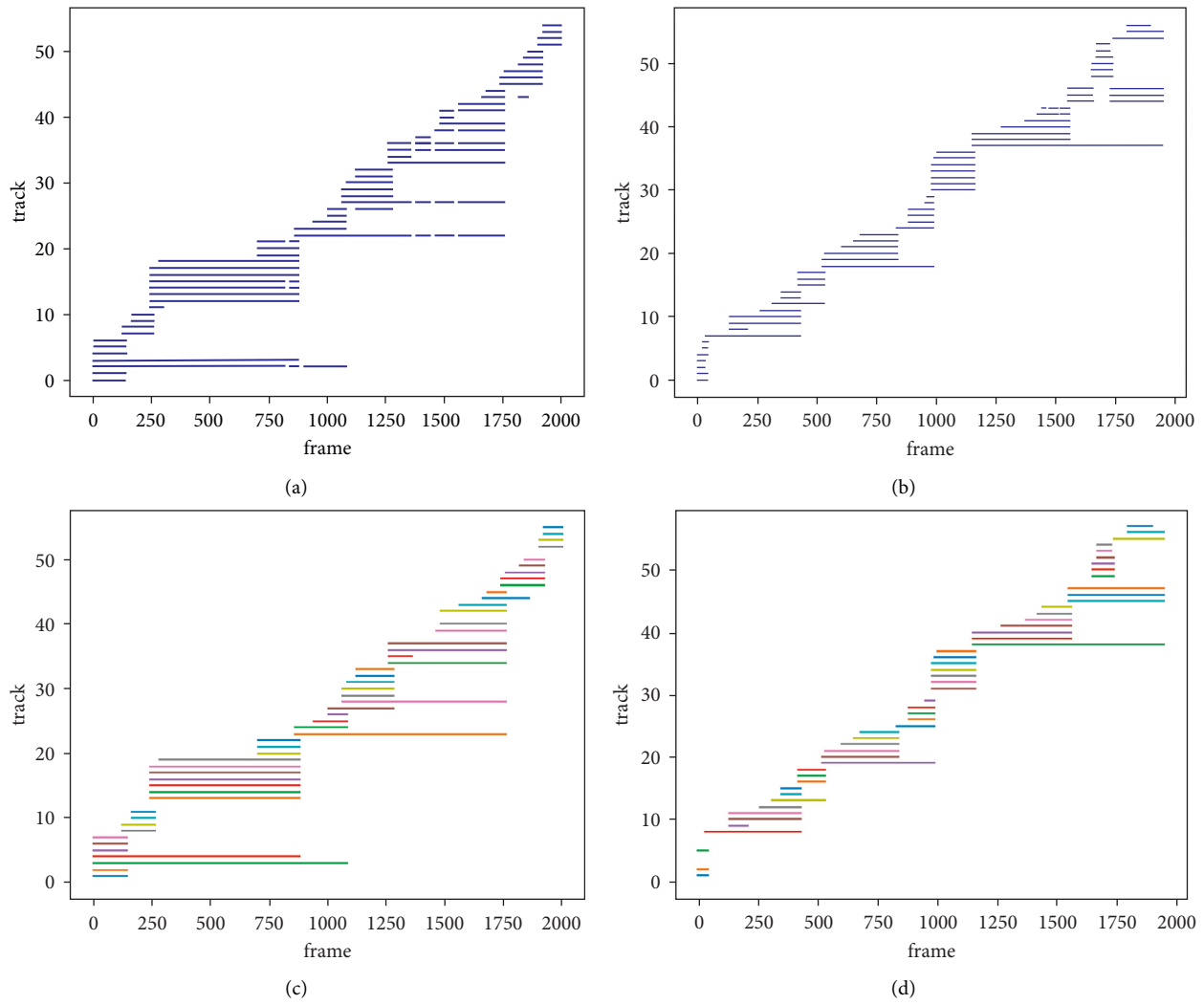


Figure 5: An example of tracking track and lifeline of content instance on a complete video timeline: (a and b) tracking trajectory; (c and d) lifeline.

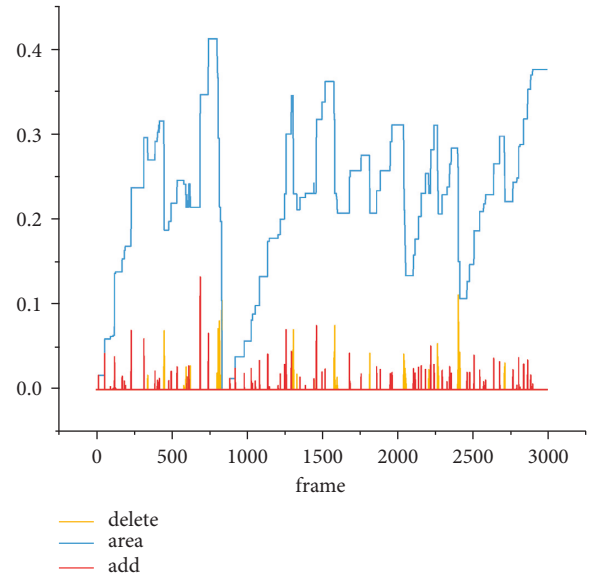


Figure 6: Normalized content addition and deletion events, as well as the size of the content area per frame.

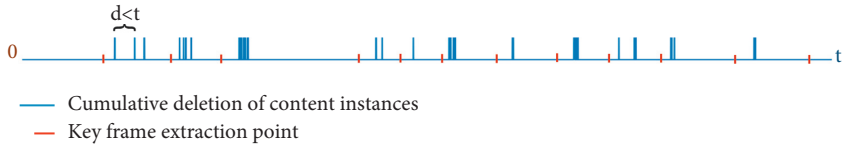


Figure 7: Video segmentation and key frame extraction.

Table 1: The information of the lecture videos.

Video number	Video duration	The number of speakers	Scene	The number of key frames
CLV01	8 m17 s	1	Whiteboard projection	3
CLV02	25 m22 s	0	Blackboard handwriting	4
CLV03	32 m37 s	1	Blackboard projection	8
CLV04	33 m11 s	1	Blackboard projection	8
CLV05	35 m28 s	1	Whiteboard projection	9
AM_01	44 m25 s	1	Whiteboard handwriting	7
AM_06	47 m20 s	1	Whiteboard handwriting	11
AM_NM_03	41 m27 s	1	Whiteboard handwriting	13

matching, i.e., one-to-one, many-to-one, and one-to-many, and uses the matrix to store the matching situation between the annotation data G and the detection result D. As shown in equation (8), the two matrices of recall and precision are denoted by σ and τ, respectively. σ, τ are matrices of $|G| \times |D|$. The probability map, adaptive threshold map, and detected bounding boxes of static frame content detection are shown in Figure 8.

$$\begin{aligned} \sigma_{ij} &= R\left(G_i, D_j\right), \\ \tau_{ij} &= P\left(G_i, D_j\right), \end{aligned} \tag{8}$$

$t_r, t_p \in [0, 1]$ are the matching judging thresholds of σ and τ, respectively, and Match() is the matching function of G and D. The rules of recall and precision calculation for a single image are shown in equation (9).

$$\begin{aligned} \mathrm{R}_{OB}\left(G, D, t_r, t_p\right) &= \frac{\sum_i \mathrm{Match}_G\left(G_i, D, t_r, t_p\right)}{|G|}, \\ \mathrm{P}_{OB}\left(G, D, t_r, t_p\right) &= \frac{\sum_j \mathrm{Match}_D\left(D_i, G, t_r, t_p\right)}{|D|}. \end{aligned} \tag{9}$$

The final recall and precision are calculated in a similar way as mAP, as shown in equations (10) and (11). The combined evaluation index F1-score is the summed average of both, as shown in equation (12).

$$\begin{aligned} \mathrm{Recall} = {} & \frac{1}{2T}\sum_{i=1}^{T} R_{OB}\left(\overline{G}, \overline{D}, i/T, t_p\right) \\ & + \frac{1}{2T}\sum_{i=1}^{T} R_{OB}\left(\overline{G}, \overline{D}, t_r, \frac{i}{T}\right), \end{aligned} \tag{10}$$

$$\begin{aligned} \mathrm{Precision} = {} & \frac{1}{2T}\sum_{i=1}^{T} P_{OB}\left(\overline{G}, \overline{D}, \frac{i}{T}, t_p\right) \\ & + \frac{1}{2T}\sum_{i=1}^{T} P_{OB}\left(\overline{G}, \overline{D}, t_r, \frac{i}{T}\right), \end{aligned} \tag{11}$$

$$F1 - \mathrm{score} = 2\frac{\mathrm{Recall} \cdot \mathrm{Precision}}{\mathrm{Recall} + \mathrm{Precision}}. \tag{12}$$

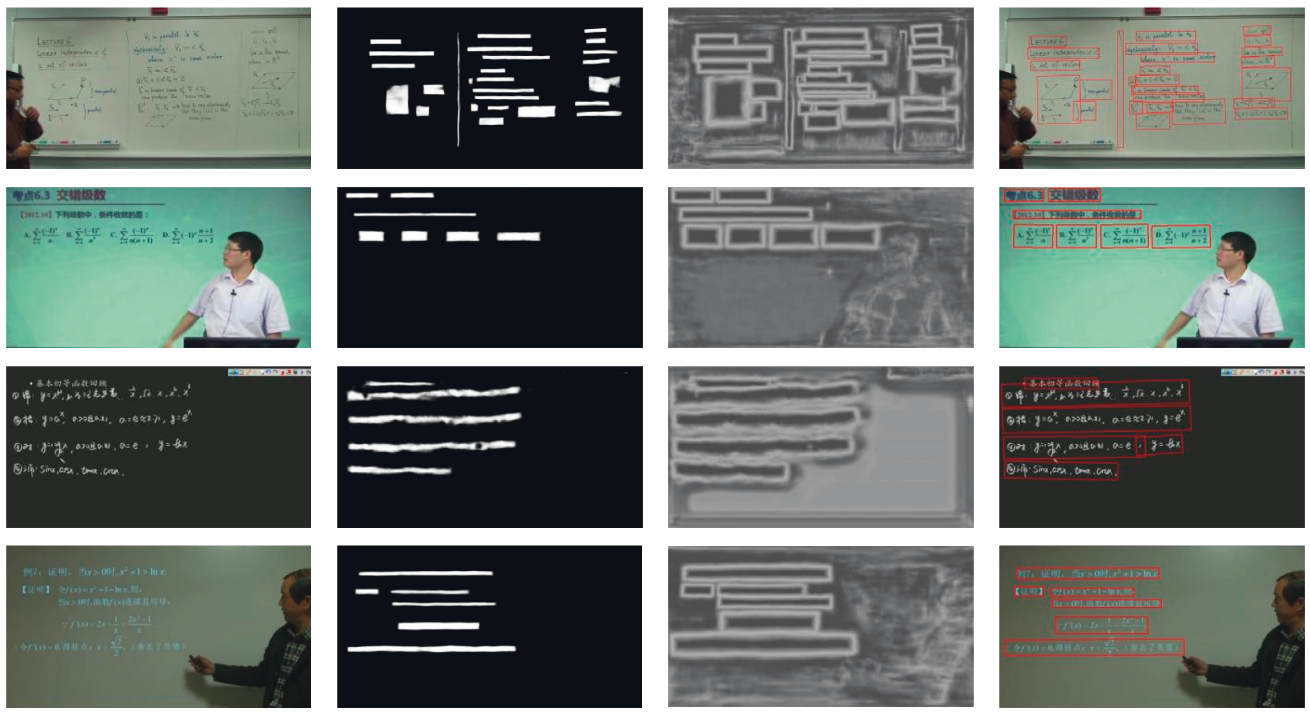
For content detection of static frames of lecture videos, the detection performance is improved when both deformable convolution and CBAM attention modules are added to the backbone network ResNet50. The precision is improved by 2.4%, the recall is improved by 4.5%, the overall index is improved by 3.5%, and the results of the ablation experiments are shown in Figure 9.

In Table 2, the content detection experiments of this paper's model are compared with the advanced text detection models PixelLink [23] and TextSnake [24], and better results are obtained by this paper's method.

3.3. Video Summary Evaluation Indexes and Experimental Results. The predicted key frames are compared with the annotated data; i.e., the summary results are matched with elements occupying the same space in approximately the same time period as the annotated data. Recall, precision, and F1-score are calculated as follows:

$$\begin{aligned} \mathrm{Recall} &= \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \\ \mathrm{Precision} &= \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \\ F1 - \mathrm{score} &= 2\frac{\mathrm{Recall} \cdot \mathrm{Precision}}{\mathrm{Recall} + \mathrm{Precision}}, \end{aligned} \tag{13}$$

where true-positive instances (TP) represent correctly predicted summary contents, and false-positive instances (FP) and false-negative instances (FN) represent incorrectly predicted (includes repeated predictions) and missing predicted contents, respectively.



(a) (b) (c) (d)

Figure 8: Visualization of content detection: (a) video static frames; (b) probability map; (c) adaptive threshold map; (d) detected bounding box results.

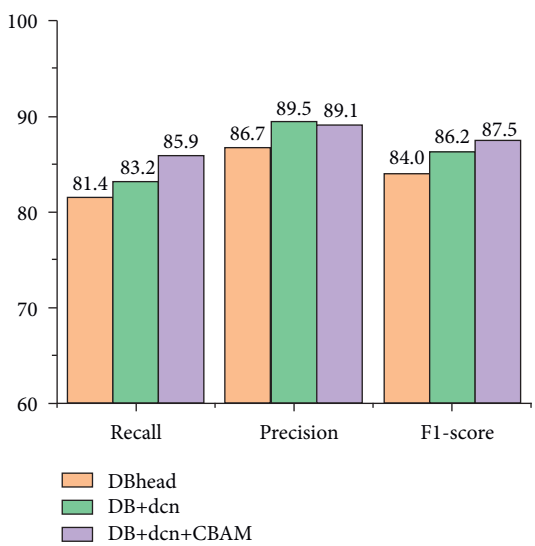


Figure 9: Results of ablation experiments.

Table 2: Comparison of different text detection models.

Models	Recall	Precision	F1-score
PixelLink	74.8	84.2	79.2
TextSnake	76.9	86.7	81.5
CBAM-DBNet	85.9	89.1	87.5

In addition, the standard deviation (SD, $\sqrt{\sum_{i=1}^{n}(K_p - K_m)^2/n}$) between the predicted key frame number K_p and the manually marked key frame number K_m is calculated to represent the compression ratio of the summary results.

Table 3: Performance of summary results on lecture videos.

Video number	The number of key frames	Recall	Precision	F1-score
CLV01	3	100	100	100
CLV02	5	84.2	72.7	78.1
CLV03	9	100	94.1	97.0
CLV04	9	87.0	87.0	87.0
CLV05	7	89.3	100	94.3
AM01	7	92.6	100	96.2
AM06	10	97.9	89.2	93.0
AM_NM_03	10	89.8	89.8	89.8
Average	1.46 (SD)	92.1	90.8	91.3

As shown in Table 3, the summary results of the method in this paper on lecture videos in various scenarios have achieved good results. The average values of precision and recall were 90.8% and 92.1%, respectively; the average composite evaluation index F1-score was 91.3%.

Among the obtained summarization experimental results, the average recall, precision, and F1-score of handwritten presentation video summarization results are 91.1%, 87.9%, and 89.3%, respectively, while the average recall, precision, and F1-score of projected presentation video summarization results are 94.1%, 95.3%, and 94.6%, respectively. The average summary performance of the method in this paper for lecture videos with handwritten presentation is lower than that of instructional videos using

projected presentation, due to the fact that handwritten content instances in lecture videos with handwritten presentation are usually irregular and the content instance detector cannot segment these tightly connected text or scribbled mathematical formulas as precisely as ground truth annotations and projected content.

Since neither the geometric position nor the appearance feature vector can distinguish the content instance with slight changes, such as the change of individual numbers in a mathematical formula, the method in this paper cannot regard the content instance with slight changes as a new content instance, which will reduce the recall rate of summary results. The method based on speaker action classification may be able to better capture these details through the speaker's action, but it is only applicable to the video of the speaker's handwriting demonstration in the whole process.

4. Conclusions

Aiming at the fact that the current detection and summarization methods based on the main visual content of educational lecture videos are often based on specific scenarios, a lecture video summarization system based on improved DBNet text detection network, Kalman filtering, and the Hungarian algorithm is proposed. The detection and summarization cover Chinese and English, handwriting, screen projection, and black and whiteboard scenes, and the summary results achieve good recall.

However, there are some shortcomings in the methodology of this paper, which will be improved in the future by the following points:

(i) Improvements will be made to the detection network to unify detection and tracking in one framework, make better use of the timing information of the video, improve detection system performance, and experiment with lightweight network structures.

(ii) Collect and label more data for more comprehensive training and analysis to further improve the robustness of the system.

(iii) The extraction and representation of the appearance features of content instances will be improved so that the improved representation can better distinguish content instances with subtle changes and improve the recall of summaries.

Data Availability

The data that support the findings of this study can be obtained from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by the Natural Science Foundation of Hebei Province of China (Grant no. F2019201329).

References

[1] L. Leiting, W. Guangli, and G. Zhouzheng, "Video summarization generation based on self-attention mechanism and random forest regression," *Computer Engineering and Applications*, vol. 58, no. 4, pp. 198–205, 2022.

[2] W. Hao and P. Li, "Video summarization algorithm based on improved fully convolutional network," *Laser & Optoelectronics Progress*, vol. 58, no. 22, pp. 415–423, 2021.

[3] H. Gharbi, S. Bahroun, and M. Massaoudi, "Key frames extraction using graph modularity clustering for efficient video summarization," in *Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing*, New Orleans, USA, March 2017.

[4] J. Wu, S. h. Zhong, J. Jiang, and Y. Yang, "A novel clustering method for static video summarization," *Multimedia Tools and Applications*, vol. 76, no. 7, pp. 9625–9641, 2017.

[5] K. Li, J. Wang, H. Wang, and Q. Dai, "Structuring lecture videos by automatic projection screen localization and analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1233–1246, 2015.

[6] K. Davila and R. Zanibbi, "Whiteboard video summarization via spatio-temporal conflict minimization," in *Proceedings of the 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, Kyoto, Japan, November 2017.

[7] M. R. Rahman, S. Shah, and J. Subhlok, "Visual summarization of lecture video segments for enhanced navigation," in *Proceedings of the 2020 IEEE International Symposium on Multimedia*, Naples, Italy, December 2020.

[8] K. Dutta, M. Mathew, and P. Krishnan, "Localizing and recognizing text in lecture videos," in *Proceedings of the 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Niagara Falls, NY, USA, August 2018.

[9] X. Zhou, C. Yao, and H. Wen, "East: an efficient and accurate scene text detector," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017.

[10] F. Xu, K. Davila, and S. Setlur, "Content extraction from lecture video via speaker action classification based on pose information," in *Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR)*, Sydney, Australia, September 2019.

[11] Z. Cao, G. Hidalgo, T. O. P. Simon, S. E. Wei, and Y. Sheikh, "OpenPose: realtime multi-person 2D pose estimation using Part Affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172–186, 2021.

[12] K. Davila, F. Xu, S. Setlur, and V. Govindaraju, "FCN-LectureNet: extractive summarization of whiteboard and chalkboard lecture videos," *IEEE Access*, vol. 9, Article ID 104469, 2021.

[13] M. Liao, Z. Wan, and C. Yao, "Real-time scene text detection with differentiable binarization," in *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, New York, NY, USA, February 2020.

[14] S. Woo, J. Park, and J. Y. Lee, "Cbam: convolutional block attention module," in *Proceedings of the 15th European Conference on Computer Vision (ECCV)*, Munich, Germany, September 2018.

[15] K. He, X. Zhang, and S. Ren, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, June 2016.

[16] J. Dai, H. Qi, and Y. Xiong, "Deformable convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, Venice, Italy, October 2017.

[17] T. Y. Lin, P. Dollar, and R. Girshick, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017.

[18] A. Bewley, Z. Ge, and L. Ott, "Simple online and realtime tracking," in *Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP)*, Phoenix, Arizona,USA, September 2016.

[19] W. Minghu, H. Yongxi, and W. Juan, "Pedestrian detection and tracking method on street based on improved YOLOv3," *Science Technology and Engineering*, vol. 21, no. 17, pp. 7230–7236, 2021.

[20] R. Jiamin, G. Ningsheng, and H. Zhenyang, "Multi-traget tracking algorithm based on YOLOV3 and kalman filter," *Computer Applications and Software*, vol. 37, no. 5, pp. 169–176, 2020.

[21] B. U. Kota, K. Davila, and A. Stone, "Automated detection of handwritten whiteboard content in lecture videos for summarization," in *Proceedings of the 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Niagara Falls, NY, USA, August 2018.

[22] C. Wolf and J. M. Jolion, "Object count/area graphs for the evaluation of object detection and segmentation algorithms," *International Journal on Document Analysis and Recognition*, vol. 8, no. 4, pp. 280–296, 2006.

[23] D. Deng, H. Liu, and X. P. Li, "Detecting scene text via instance segmentation," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, San Francisco, CA, USA, February 2018.

[24] S. Long, J. Ruan, and W. T. Zhang, "A flexible representation for detecting text of arbitrary shapes," in *Proceedings of the 15th European Conference on Computer Vision (ECCV)*, Munich, Germany, September 2018.