ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-6654-0540-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICASSP43922.2022.9746742

A LIGHT WEIGHT MODEL FOR VIDEO SHOT OCCLUSION DETECTION

Junhua Liao[1], *Haihan Duan*[2,3], *Wanbin Zhao*[1], *Yanbing Yang*[1,4] *and Liangyin Chen*[1,4,†]

[1] College of Computer Science, Sichuan University, Chengdu, China
[2] The Chinese University of Hong Kong, Shenzhen, China
[3] Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen, China
[4] The Institute for Industrial Internet Research, Sichuan University, Chengdu, China

ABSTRACT

The popularity of video social platforms (TikTok, etc.) shows that video is a popular information carrier at present. However, shot occlusion frequently occurs when people are shooting videos to record information. Since the shot occlusion seriously affects the viewers' experience, the video editors need to find and delete such segments from the video material during post-processing. However, finding the shot occlusion from the video is a time-consuming and laborious task. To reduce the workload of editors, previous researchers proposed a shot occlusion detection algorithm using deep learning technology, which has promotion space in both recognition accuracy and computational efficiency. In this paper, we propose a neural network module, named SAT module, which can effectively extract spatio-temporal information with fewer parameters. We apply SAT module to construct a novel occlusion detection model, and improve the existing occlusion detection loss function for model training. The experimental results on the public dataset show that our method achieves the state-of-the-art performance of 88.25% accuracy and FPS of 130 with the least parameters. Code and models will be available at `https://github.com/Junhua-Liao/ICASSP22-OcclusionDetection`.

***Index Terms*—** Occlusion Detection, Automatic Video Editing, Human-centered Computing, Deep Learning

1. INTRODUCTION

The popularity of smartphones has significantly reduced the cost of video shooting, so the video becomes the first choice when people need to record more valuable information. However, the original video material usually needs to be post-produced by professional video editors to facilitate retention and dissemination. The massive video materials bring a great workload to video editors in this era of information explosion.

Fortunately, automatic video editing technology has brought the dawn to mitigate the workload of editors. Due to the complexity and diversity of video materials, different kinds of videos have different editing rules, so the existing automatic editing technologies are generally limited to a small field, such as multi-party conversations [1], school concerts [2], social gatherings [3], and dialogue-driven scenes [4]. However, affected by the shooting equipment, shooting method, shooting environment, etc., the original video materials may have low-quality segments resulting in unsatisfactory visual effects. Therefore, no matter what type of video, editing usually involves the procedure of eliminating low-quality video segments [5, 6]. The common causes of low-quality video segments are shot blurriness, shake, and occlusion. Among them, shot occlusion will highly influence the viewers' experience, as shown in Fig.1.

Fig. 1. Examples of shot occlusion in different situations.

Shot occlusion refers to the phenomenon that inappropriate objects break into the picture of the shot and lead to the protagonist being occluded during shooting. Recently, occlusion has attracted extensive attention in a series of related tasks [7, 8, 9, 10, 11]. Among them, Liao et al. [12] proposed a shot occlusion detection method to assist editors to find the video segments with shot occlusion. However, this method based on 3D convolution requires heavy computational resources and the recognition accuracy still has space for improvement. Therefore, it is worth proposing a lightweight model, which can reduce the number of model parameters and also improve the performance of shot occlusion detection.

In this paper, we propose a new occlusion detection method to improve the defects of previous work. The main

This work is supported in part by the National Natural Science Foundation of China under Grant 62072319; in part by the Sichuan Science and Technology Program under Grant 2019JDTD0001.

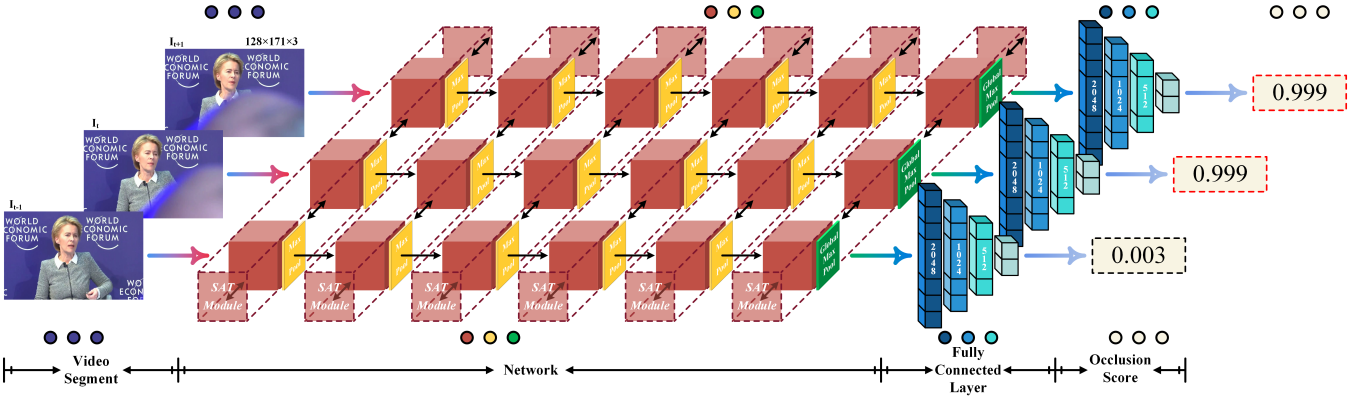

Fig. 2. The architecture of the proposed shot occlusion detection model.

contributions of this paper can be summarized as follows:

(1) We design a novel neural network module, named SAT, to extract spatio-temporal information instead of 3D convolution, and construct a new high-performance video shot occlusion detection framework based on this module.

(2) We improve the existing occlusion detection loss function to more reasonably assign weights to occlusion frames, which significantly increases the accuracy of recognition.

(3) The extensive experiments show that our proposed method outperforms the state-of-the-art methods, especially in terms of the number of module parameters.

2. METHOD

2.1. Occlusion Detection Algorithm

This section will introduce our shot occlusion detection method in detail. Fig.2 illustrates the architecture of the proposed shot occlusion detection model. Given the input video sequences, we first resize the video frames with different resolutions to a specific size and divide them into a fixed number of segments. Next, we send the processed data to the network model constructed by the SAT modules for spatio-temporal feature extraction. When extracting spatio-temporal information, this module will only change the channel dimension rather than the spatio-temporal dimension. The changes of channel dimensions are shown in Table 1. More information about the SAT module will be introduced in Sec. 2.2. In the network, except for the last SAT module, all SAT modules are followed by a 2D pooling layer with kernel size and stride of 2, which performs max-pooling of features. Then, we use the global max pooling to convert the extracted features into vectors of specific dimensions and send them to the fully connected layer for processing. The fully connected layer is composed of three layers with dimensions of 1024, 512, 2. The dropout layers between each fully connected layer are used to prevent over-fitting. Finally, the softmax function uses the output of the fully connected layer to calculate the occlusion score. The frames with occlusion scores greater than or equal to the threshold are considered to have shot occlusion, where the occlusion threshold is 0.5 in this paper. Because shot occlusion is a time series process, such an architecture design can combine the information between the previous and next frames to make accurate predictions.

Table 1. Channel number of SAT module inputs and outputs.

Channel	1st	2nd	3rd	4th	5th	6th
C_{input}	3	64	128	256	512	1024
C_{output}	64	128	256	512	1024	2048

2.2. SAT Module

3D convolution is usually the first choice for extracting spatio-temporal features [13], but this method consumes more computational resources due to a large number of parameters. Therefore, some researchers decompose 3D convolution into 1D convolution and 2D convolution to process temporal and spatial information respectively to reduce model parameters [14, 15]. Inspired by this, we propose a new spatio-temporal feature processing module SAT, as shown in Fig.3. The input dimension of this module is $T \times H \times W \times C_i$ and the output dimension is $T \times H \times W \times C_o$, where T, H, W, C_i, and C_o are the temporal, height, width, input channel number, and output channel number, respectively. It can be seen that the SAT module does not change the temporal, height, and width of features when extracting spatio-temporal information. This design is helpful to the fusion of spatial and temporal features. The SAT module is unique since it internally consists of four parallel information extraction paths. (1) All paths use convolution with a kernel size of 1 to process the input features to obtain transition features with a size of $T \times H \times W \times \frac{C_o}{8}$. (2) Path I and path III use 2D convolution and 1D convolution with a kernel size of 3 to extract spatial and temporal features, respectively. The size of the convolution kernel for spatial and temporal information extraction

in path II and path IV is 5. This design of convolution using receptive fields of different sizes is conducive to extracting more abundant information. (3) The features extracted from the four paths are concatenated in the channel dimension and output. Experiments show that this lightweight module can extract effective information from spatio-temporal features.

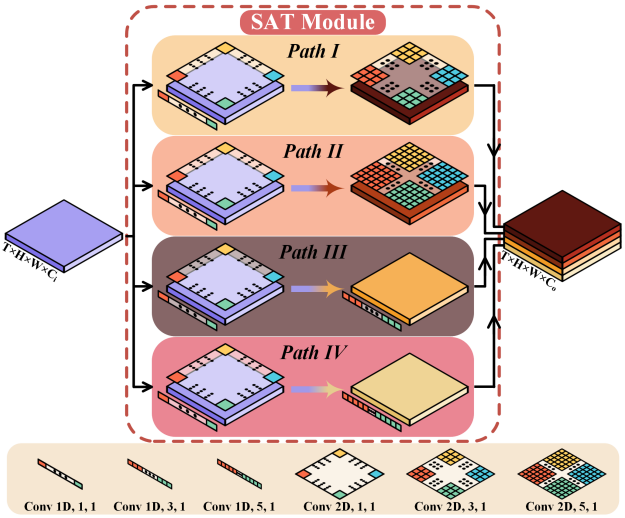

Fig. 3. Structure of the SAT module. The $Conv\ xD, k, s$ represents xD convolution with kernel size k and stride s.

2.3. Occlusion Loss Function

In order to increase the sensitivity to shot occlusion, the previous work [12] designs a loss function to weigh the video frame according to the percentage of the occlusion area in the frame. This function will assign larger weights to large occlusions, and not pay much attention to small occlusions. However, this way of assigning weights is not reasonable. For example, the small occluder in Fig.1 can also have a great impact on viewing. The original occlusion loss function does not pay more attention to these small occlusions which cause serious impact, because it only values the large-area occlusion. Therefore, we improve the original occlusion loss function to solve this problem. The specific steps of this loss function are as follows.

Firstly, we calculate the percentage $\mathcal{P}_{occlusion}$ of the occlusion area in the frame.

$$\mathcal{P}_{occlusion} = \frac{\mathcal{A}_{occlusion}}{\mathcal{A}_{frame}} \tag{1}$$

Where $\mathcal{A}_{occlusion}$ is the area of occlusion and $\mathcal{A}_{frame}$ is the area of the frame.

Secondly, we calculate the maximum $\mathcal{P}_{occlusion}$ in the video, which is called $\mathcal{M}_{occlusion}$.

$$\mathcal{M}_{occlusion} = max(\mathcal{P}_{occlusion} \in \mathcal{V}_i) \tag{2}$$

Where $\mathcal{V}_i$ represents the entire sequence of the i-th video.

Thirdly, we calculate the occlusion ratio $\mathcal{R}_{occlusion}$, which is calculated as follows.

$$\mathcal{R}_{occlusion} = \frac{2 - (\mathcal{P}_{occlusion} + \frac{\mathcal{P}_{occlusion}}{\mathcal{M}_{occlusion}})}{2\beta} \tag{3}$$

Where β is set to 10 as an equilibrium coefficient empirically.

Finally, our proposed occlusion loss function $\mathcal{L}_{occlusion}$ calculates weights according to $\mathcal{R}_{occlusion}$ and assigns the weights to the corresponding frames.

$$\mathcal{L}_{occlusion} = -e^{-\mathcal{R}_{occlusion}}(t_j^i \log(p_j^i) + (1 - t_j^i) \log(1 - p_j^i)) \tag{4}$$

where t and p represent the tag and prediction results respectively, and ^i_j represents the j-th frame of the i-th video.

It can be seen from formula (3) that, compared with the original loss function, our loss function additionally considers the ratio of the current occlusion area to the maximum occlusion area in the video sequence when calculating the weight. Shot occlusion lasts from the occluder intruding into the shot until the occluder leaves the shot. In this process, regardless of the size of the occluder, when its proportion in the shot reaches the maximum, the bad visual effect will be the greatest. Therefore, this strategy allows the model to pay attention to those small occlusions which have a serious impact on the viewing. Theoretically, this occlusion loss function is more reasonable for the assignment of the weight.

3. EXPERIMENTS

3.1. Data and Criterion

We validated our shot occlusion detection method on the public data set [12]. This dataset is the first large-scale public dataset for the video shot occlusion detection task. It consists of 1000 videos from seven scenes in the real world, and is officially divided as the training set and testing set according to the ratio of 8:2. Each video is annotated frame by frame by practitioners with basic computer vision knowledge.

We use the model parameters, frame-level accuracy, frames per second (FPS), receiver operating characteristic (ROC) curve, and its corresponding area under the ROC curve (AUC) to evaluate the performance of the shot occlusion detection method.

3.2. Implementation

The input of the networks is a continuous eight-frame video sequence with a size of 128 × 171 pixels. The networks were trained to utilize the Stochastic Gradient Descent (SGD) in 50 epochs of training, where the momentum was 0.9 and weight decay was 0.0005. And the learning rate is set as 0.0005 with a decay rate of 0.5 every 10 epochs. All models are implemented with PyTorch [16] and experimented on an NVIDIA GTX 1080Ti GPU (11GB).

3.3. Results and Comparison

Since video shot occlusion detection is a novel research task, there are few related types of research. In order to demonstrate the performance of our proposed shot occlusion detection method, we choose the state-of-the-art method of this task [12], two state-of-the-art models of decoupled 3D convolution [14, 15], three classical classification models [17, 18, 19], and four state-of-the-art occlusion detection methods [7, 8, 9, 10] in other tasks for comparison.

Table 2 shows the comparison results of our proposed shot occlusion detection algorithm and other methods in terms of model parameters, testing set accuracy and FPS. Fig. 4 shows the ROC curves and AUC values of various methods. The experimental results show that our shot occlusion detection method achieves the best performance of 88.25% accuracy, AUC value of 0.95, and FPS 130 with the smallest model. Compared with the state-of-the-art method [12], our method not only reduces the number of parameters by more than 5 times, but also improves the accuracy by 5.55%. This indicates that our method has made a major breakthrough in the task of shot occlusion detection.

Table 2. Performance comparison with the state-of-the-art methods on public shot occlusion detection dataset.

Method	Parameters	Accuracy	FPS
VGG-19 [17]	139.59M	68.85%	70
ResNet-101 [18]	42.50M	61.06%	83
DenseNet-169 [19]	12.49M	65.56%	95
R(2+1)D [14]	33.18M	59.10%	99
P3D [15]	24.93M	74.09%	120
Hou et al. [7]	23.51M	42.66%	60
Zhu et al. [8]	15.76M	73.17%	61
Lazarow et al. [9]	64.66M	62.26%	32
Chi et al. [10]	40.78M	50.43%	33
Liao et al. [12]	59.64M	82.70%	106
Our Method	**11.37M**	87.03%	**130**
Our Method+$\mathcal{L}_{occlusion}$	**11.37M**	**88.25%**	**130**

According to the results, we can find that the accuracy of the classical classification models [17, 18, 19] fail to reach 70%. It may be that they did not introduce temporal information for prediction like the SAT module. R(2+1)D [14] and P3D [15] reduce the model parameters by decoupling 3D convolution, but also cause the loss of spatio-temporal information, resulting in unsatisfactory recognition accuracy. Due to the differences between tasks, the state-of-the-art occlusion detection methods [7, 8, 9, 10] in other scenarios do not perform well after migrating to this task. As the state-of-the-art method for video shot occlusion detection, Liao et al. [12] has the best performance in the comparison methods. This method uses 3D convolution for spatio-temporal feature extraction, so it is computationally expensive. The SAT module greatly reduces the number of model parameters by decomposing 3D convolution, and uses different sizes of convolution kernels to extract richer information when processing spatio-temporal features. The experimental results show that our proposed shot occlusion detection method is simple but efficient. In addition, we improve the existing occlusion loss function, which can reasonably allocate the weight for videos with different occlusion sizes. By replacing the original loss function [12] with our loss function for training, the shot occlusion detection accuracy of our method increases from 87.03% to 88.25%. The accuracy is improved by 1.22%, which proves the effectiveness of this novel loss function.

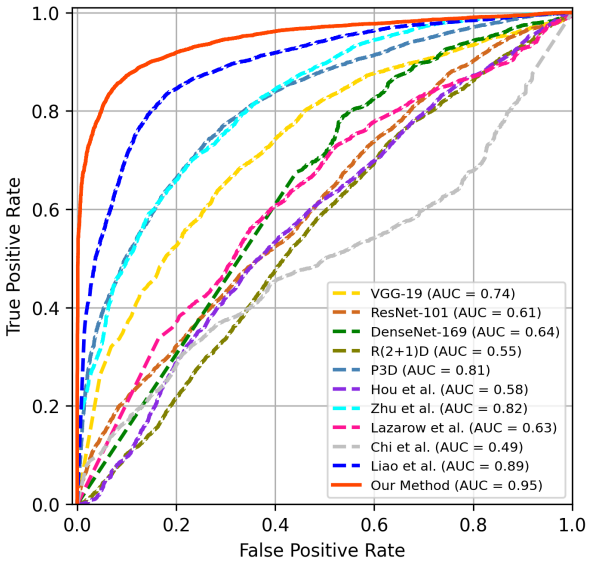

Fig. 4. ROC curves for shot occlusion detection methods.

4. CONCLUSIONS

In this paper, we proposed a simple but efficient shot occlusion detection method based on the SAT module. The SAT module decomposes the 3D convolution into 1D convolution and 2D convolution with multi-scale receptive fields to process the temporal and spatial characteristics respectively and simultaneously, so as to obtain more information for occlusion detection while reducing the number of parameters. Particularly, we proposed a new occlusion loss function, which can more legitimately allocate the weight of occluded frames to achieve better performance. Plentiful experimental results on the public dataset show that the proposed method outperforms the state-of-the-art methods on the video shot occlusion detection task. As future work, we plan to deploy this model on photographic equipment for real-time shot occlusion detection. This research is very valuable and challenging, because it can not only remind the cameraman to adjust the position of the equipment in time, but also obtain the position of the occlusion segments at the end of the shooting.

5. REFERENCES

[1] Yoshinao Takemae, Kazuhiro Otsuka, and Naoki Mukawa, "Video cut editing rule based on participants' gaze in multiparty conversation," in *Proceedings of the eleventh ACM international conference on Multimedia*, 2003, pp. 303–306.

[2] Rodrigo Laiola Guimaraes, Pablo Cesar, Dick CA Bulterman, Vilmos Zsombori, and Ian Kegel, "Creating personalized memories from social events: community-based support for multi-camera recordings of school concerts," in *Proceedings of the 19th ACM international conference on Multimedia*, 2011, pp. 303–312.

[3] Vilmos Zsombori, Michael Frantzis, Rodrigo Laiola Guimaraes, Marian Florin Ursu, Pablo Cesar, Ian Kegel, Roland Craigie, and Dick CA Bulterman, "Automatic generation of video narratives from shared ugc," in *Proceedings of the 22nd ACM conference on Hypertext and hypermedia*, 2011, pp. 325–334.

[4] Mackenzie Leake, Abe Davis, Anh Truong, and Maneesh Agrawala, "Computational video editing for dialogue-driven scenes.," *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 130–1, 2017.

[5] Anh Truong, Floraine Berthouzoz, Wilmot Li, and Maneesh Agrawala, "Quickcut: An interactive tool for editing narrated video," in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, 2016, pp. 497–507.

[6] Miao Wang, Guo-Wei Yang, Shi-Min Hu, Shing-Tung Yau, and Ariel Shamir, "Write-a-video: computational video montage from themed text," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–13, 2019.

[7] Ruibing Hou, Bingpeng Ma, Hong Chang, Xinqian Gu, Shiguang Shan, and Xilin Chen, "Vrstc: Occlusion-free video person re-identification," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 7176–7185.

[8] Meilu Zhu, Daming Shi, Mingjie Zheng, and Muhammad Sadiq, "Robust facial landmark detection via occlusion-adaptive deep networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3486–3496.

[9] Justin Lazarow, Kwonjoon Lee, Kunyu Shi, and Zhuowen Tu, "Learning instance occlusion for panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10720–10729.

[10] Cheng Chi, Shifeng Zhang, Junliang Xing, Zhen Lei, Stan Z Li, and Xudong Zou, "Pedhunter: Occlusion robust pedestrian detector in crowded scenes," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, vol. 34, pp. 10639–10646.

[11] Yifeng Chen, Guangchen Lin, Songyuan Li, Omar Bourahla, Yiming Wu, Fangfang Wang, Junyi Feng, Mingliang Xu, and Xi Li, "Banet: Bidirectional aggregation network with occlusion handling for panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3793–3802.

[12] Junhua Liao, Haihan Duan, Xin Li, Haoran Xu, Yanbing Yang, Wei Cai, Yanru Chen, and Liangyin Chen, "Occlusion detection for automatic video editing," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, p. 2255–2263.

[13] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[14] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 6450–6459.

[15] Zhaofan Qiu, Ting Yao, and Tao Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5533–5541.

[16] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al., "Pytorch: An imperative style, high-performance deep learning library," in *Advances in neural information processing systems*, 2019, pp. 8026–8037.

[17] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[19] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269.