Shot Boundary Detection Through Multi-stage Deep Convolution Neural Network

Tingting Wang[1], Na Feng[1], Junqing Yu[1](✉), Yunfeng He[1], Yangliu Hu[1], and Yi-Ping Phoebe Chen[2]

[1] School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China
{fengna,yjqing}@hust.edu.cn

[2] Department of Computer Science and Information Technology, La Trobe University, Melbourne, VIC 3086, Australia
phoebe.chen@latrobe.edu.au

Abstract. Fast and accurate shot segmentation is very important for content-based video analysis. However, existing solutions have not yet achieved the ideal balance of speed and accuracy. In this paper, we propose a multi-stage shot boundary detection framework based on deep CNN for shot segmentation tasks. The process is composed of three stages, which are respectively for candidate boundary detection, abrupt detection and gradual transition detection. At each stage, deep CNN is used to extract image features, which overcomes the disadvantages of hand-craft feature-based methods such as poor scalability and complex calculation. Besides, we also set a variety of constraints to filter as many non-boundaries as possible to improve the processing speed of the model. In gradual transition detection, we introduce a scheme that can infer the gradual position by computing the probability signals of the start, mid and end of the gradual transition. We conduct experiments on ClipShots and the experimental results show that the proposed model achieves better performance on abrupt and gradual transition detection.

Keywords: Shot boundary detection · Convolution neural network · Abrupt transition · Gradual transition

1 Introduction

The research of shot boundary detection has been carried on for many years. However, due to the diversity of the gradual transition and the interference in the video, such as sudden light changes, rapid movement of objects or cameras, etc., the problem of shot boundary detection is still not well solved. Most methods use the shallow visual features designed by prior knowledge and lack the ability to describe high-level semantic information. Fast detection methods usually analyze spatial information and adopt simple identification mechanisms, and the results

J. Lokoč et al. (Eds.): MMM 2021, LNCS 12572, pp. 456–468, 2021.
https://doi.org/10.1007/978-3-030-67832-6_37

are not as good as expected. To improve the accuracy, some methods adopt more complex feature combination and identification mechanism, resulting in high computational cost and slower speed. Since shot segmentation is used as a preprocessing step for video content analysis, it is important to simultaneously improve the accuracy and speed of shot boundary detection.

In recent years, thanks to the development of Graphics Processing Unit (GPU) and large-scale datasets, deep learning has achieved major breakthroughs in speed and efficiency on image and video analysis issues. Compared with manually designed features, its ability to automatically learn and extract high-level semantic feature expression can better reflect the diversity of data. However, there are few studies on applying CNN to shot segmentation.

In this paper, we propose a CNN-based multi-stage shot boundary detection framework (SSBD). It is divided into three stages. The first stage is to generate candidate boundaries where the shot transition may occur. CNN is used as the feature extractor of the frame and then the most non-boundaries are quickly excluded by calculating the difference between adjacent frames. In the second stage, 3D CNN is used to further identify the abrupts, gradual transitions and non-boundaries. Threshold mechanism is adopted to obtain candidate gradual frames from the latter two. The third stage is to detect gradual transitions. We still adopt 3D CNN to predict the probability that each candidate frame belongs to the start, mid and end of the transition, and then the position can be determined by strong peaks of these three probability signals.

The contributions of our work can be summarized as follows:

- We propose a shot boundary detection framework and conduct experiments on ClipShots. The results show that the proposed model performs better than others.
- We put forward a variety of constraints in the process of shot boundary detection, which can quickly and accurately filter out non-shot boundaries to improve the processing speed and reduce calculations.
- We introduce a scheme in the gradual transition detection, which calculates the probability signals of the start, mid and end of the transition, and then the position can be determined according to the strong peaks of these signals.

2 Related Works

Traditionally, most shot boundary detection methods mainly rely on well-designed hand-crafted features. The basic idea is to achieve shot segmentation by finding the changing rule of the difference between frames at the shot boundary. These methods usually includes three steps: visual content representation (feature extraction), construction of continuous signals (similarity measure), and the shot boundary classification of the continuous signals (shot boundary identification). The features used by these methods include color histograms [1,2], edges [3], mutual information and entropy [4], wavelet representation [5], speeded up robust features (SURF) [6], motion information [5,7,8] and many other manual features [9–11]. The threshold mechanism [4,12–14] has been widely used in the decision-making stage, but recently most researchers employ statistical learning algorithms to identify shot boundaries.

In order to eliminate the interference caused by illumination and camera or object movement, some methods tend to make use of complex features and continuous calculations but cannot achieve real-time analysis. As the basis of high-level video content analysis, efficient detection of shot boundaries is also important. [2] proposes a method for fast detection based on singular value decomposition and pattern matching. [6] employs SURF descriptors and HSV histogram to describe the visual feature of the image, and detect abrupt and gradual transitions by calculating the similarity between adjacent frames. In addition, the paper also proposes a GPU-based computing framework to achieve real-time analysis. [15] proposes a multi-modal visual features-based framework, which uses the discontinuity signal calculated based on SURF and RGB histogram. The above methods only use the spatial features of the image, and the processing speed is very fast but at the cost of the accuracy.

Encouraged by the successful application of deep learning on visual tasks, researchers have begun to use deep learning to achieve shot boundary detection in the past two years, but there are still few related works. [16] proposes a method based on interpretable labels learned by CNN. It uses a similar mechanism as in [2] to eliminate non-boundary frames, and then adopt the pixel-wise difference and the adaptive threshold-based method to detect abrupt. For gradual transition, it uses CNN to get labels of the previous and next frames of one candidate segment and analyze the relationships between those labels to judge if the segment has gradual transition in it. [17] and [18] apply 3D CNN to identify abrupt and gradual transitions. [19] introduces a cascade framework that can achieve rapid and accurate shot segmentation. It first extracts CNN features of the image to filter non-boundaries and then uses 2D CNN and 3D CNN to identify the abrupt and gradual transitions. Although these methods have made improvements on shot boundary detection, there are still some problems, such as the inability to accurately localize the boundaries, the lack of tolerance for variable shot lengths, etc.

3 Methodology

This part describes the proposed method in detail. Firstly, we use CNN to extract the spatial feature of each frame in the video and then detect possible shot boundaries by calculating the difference between adjacent frames. Secondly, 3D CNN is used to extract the spatio-temporal features of the candidate boundary frames and their neighbors to identify the abrupt transitions. At the same time, the probability threshold is used to generate candidate gradual frames. Finally, predicting the probability that each candidate gradual frame belongs to the start, mid and end of the transition, and the position can be derived from strong peaks of these three probability signals. The pipeline is illustrated in Fig. 1.

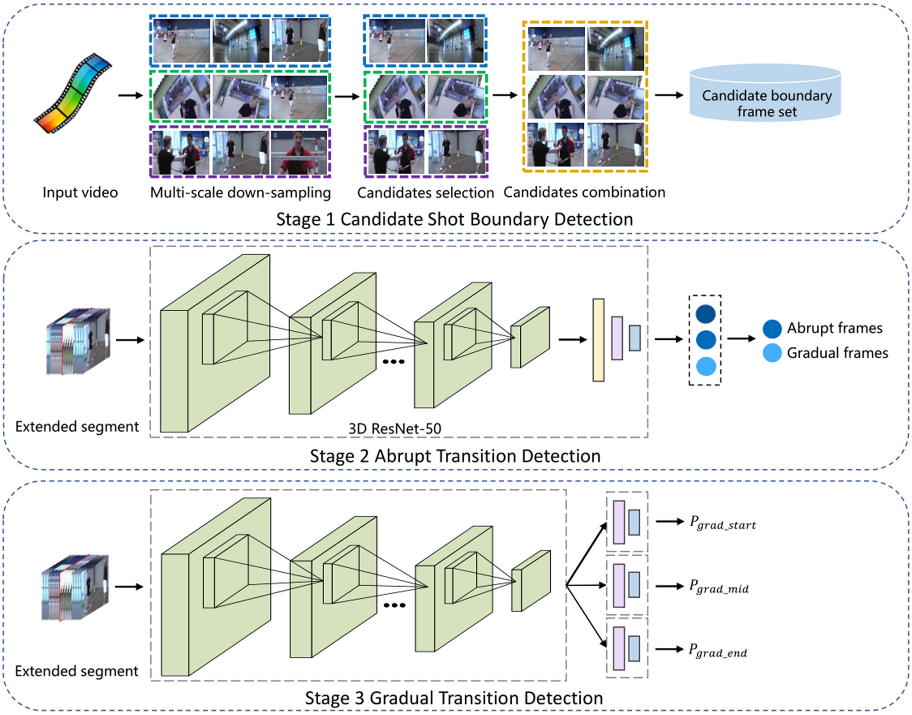


Fig. 1. The pipeline of SSBD. It consists of three stages. In the first stage, multiple scales are used to sample the video and then pick out potential shot boundary frames by calculating the difference between frames. In the second stage, the candidate frames are expanded into segments and 3D ResNet-50 is used to predict abrupts and candidate gradual frames. In the third stage, we use 3D CNN to predict the probability that each candidate frame belongs to the start, mid, and end of the gradual transition, and then construct the probability signal function to infer the position of the transition.

3.1 Candidate Shot Boundary Detection

Shot is a group of images continuously captured by the same camera. The visual content of the image in the same shot is continuous in time and space but inconsistent in different shots. We adopt the visual difference between two consecutive frames as the measure of visual continuity, it can be seen that the difference maintains a stable change rate within the same shot, but changes significantly when the shot transition occurs. Therefore, we compare the difference with the predefined threshold to preserve the shot boundaries. If the difference between two consecutive frames is greater than the threshold, it can be considered that a shot transition occurs. The specific steps are as follows:

1. Use CNN to extract the spatial feature of each frame in the video sequence and denote it by F_i.

2. Calculate the difference d_i between the i-th and the i+1-th frames by the following equation

$$d_i = 1 - \frac{< F_i, F_{i+1} >}{\|F_i\| \, \|F_{i+1}\|} \tag{1}$$

   where $<F_i,F_{i+1}>$ represents the dot product of F_i and F_{i+1}, $\|F_i\|$ represents the L2 norm of F_i.
3. Calculate the mean value μ_G of the difference of all frames in the video sequence.
4. For the i-th frame, if it satisfies $(d_i > \lambda d_{i-1} \cup d_i > \lambda d_{i+1}) \cap d_i > \gamma \mu_G$, it is regarded as a candidate boundary frame. λ specifies the minimum change rate of visual content when shot transition occurs, γ and μ_G constitute the global static threshold of the difference between frames.

Since the length of the gradual transition varies greatly, we use multiple temporal scales to downsample the video and then merge the candidate frames obtained at different scales. When two candidate frames at different scales are very close (within five frames), only the candidate frame at the lower scale is retained. In the experiment, we use scales of 1, 2, 4, 8, 16, and 32. In addition, we consider VGG-16, ResNet-50 and SqueezeNet as feature extraction networks and use the output of high layers as feature representations. Specifically, the fc6 of VGG16, the pool5 of ResNet-50 and the pool10 of SqueezeNet.

3.2 Abrupt Detection

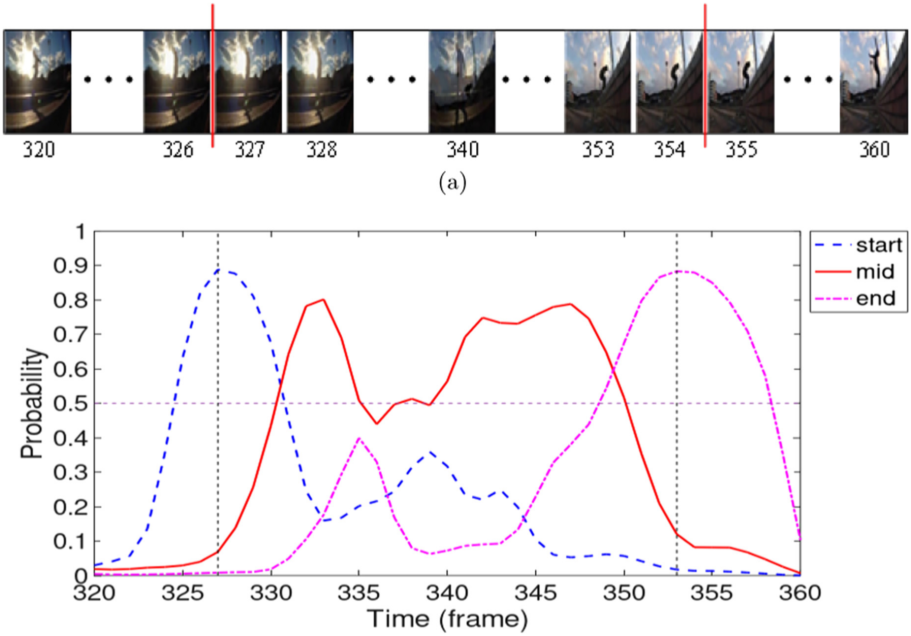
The input of the abrupt detection model is a set of continuous frames centered on the candidate frame. For the candidate frame x, it is expanded 7 and 8 frames forward and backward respectively to form a segment with a length of 16. When x is the first or last frame of the video, that is, $x - 7$ is less than 0 or $x + 8$ is greater than the total number of video frames, it needs to be looped multiple times to form a 16-frame segment. After that, we choose 3D ResNet-50 as the classification network and output the probability that the frame is abrupt, gradual and non-boundary. To prevent some negative samples from being predicted as abrupts, simple post-processing is performed on all abrupt frames:

1. For abrupt frame x and its neighbor $x + 1$, calculate the HSV histograms H_x and H_{x+1}, where H is set to 18, S is set to 16 and V is set to 16.
2. Calculate the Bhattacharyya distance d between H_x and H_{x+1} by the following equation

$$d(H_x, H_{x+1}) = \sqrt{1 - \frac{1}{\sqrt{\bar{H}_x \bar{H}_{x+1} N^2}} \sum_I \sqrt{H_x(I) \cdot H_{x+1}(I)}} \tag{2}$$

3. Compare d with the threshold T. If $d < T$, it is considered that there is no abrupt at x. Experiments show that the best result is obtained when T is set to 0.36.

Although the abrupt detection network also outputs predictions of gradual transitions and non-shot boundaries, the lack of gradual transition training samples may lead to inaccurate recognition. Therefore, in addition to those boundaries predicted to be gradual transitions, the non-boundaries whose gradual transition probability is greater than or equal to the threshold p are also retained as the potential gradual transitions. They are all the inputs for the next stage. In the experiment, p is set to 0.1.



(a)

(b)

Fig. 2. Example of gradual transition prediction. (a) displays a gradual transition in [327,353], (b) shows three state probability signals of the gradual transition.

3.3 Gradual Transition Detection

This stage aims to locate the gradual transitions in the video. Inspired by [20], the model we build identifies three gradual transition states: start (the first frame of the transition), end (the last frame of the transition), and mid (any frame between the first and last of the transition). After obtaining all candidate gradual frames, we first expand them into candidate segments and then use 3D CNN to compute the probability that each frame in the segments belongs to the above three states. Finally, gradual transitions can be determined based on these three signals.

For a given candidate gradual frame x, it is expanded n frames forward and backward respectively to form a candidate segment with a length of 2n+1.

Since the segment cannot overlap the abrupt, it should not span the abrupt closest to x. Let N_{total} be the total number of video frames. $C_{left}(x)$ represents the abrupt frame closest to the left of x, if the value is -1, there is no abrupt. $C_{right}(x)$ represents the abrupt frame closest to the right of x, if the value is $N_{total} - 1$, there is no abrupt. L_{min} represents the minimum length of the shot. In this paper, we use the last frame of the previous shot to represent the abrupt. Thus, the candidate interval of x is $(\max(C_{left}(x)+1+L_{min},\ x-n+1),\ \min(x+n-1,\ C_{right}(x)-L_{min}))$. In the experiment, n takes 25 and L_{min} is set to 1.

Algorithm 1. Process of gradual transition detection

```
Input: Smooth probability signal $g_{start}(t)$, $g_{mid}(t)$ and $g_{end}(t)$;
Output: Gradual transitions;
Find all the maximum points in $g_{mid}(t)$, each is denoted by $m_i$;
for $i$ = 1,2,3,... do
    if $g_{mid}(m_i) \geq Th_{mid}$ then
        for $j = m_i$ to $m_i - separation$ do
            find the strong peaks of $g_{start}(t)$ (the probability is not less than $Th_{s,e}$)
            if there is a strong peak of $g_{end}(t)$ between two consecutive strong peaks then
                end the scan and keep the strong peak points on the right
            end if
        end for
        if $g_{start}(t)$ has one or more strong peak points then
            choose the point that maximizes the value of $g_{start}(t)$ as the start of the gradual transition denoted by $s_i$
            for $k = m_i$ to $m_i + separation$ do
                find the strong peaks of $g_{end}(t)$ (the probability is not less than $Th_{s,e}$)
                if there is a strong peak of $g_{start}(t)$ between two consecutive strong peaks then
                    end the scan and keep the strong peak points on the left
                end if
            end for
            if $g_{end}(t)$ has one or more strong peak points then
                choose the point that maximizes the value of $g_{end}(t)$ as the end of the gradual transition denoted by $e_i$. The gradual transition interval is ($s_i$, $e_i$, $g_{mid}(m_i)$)
            end if
        end if
    end if
end for
Perform NMS on all gradual transitions. When multiple gradual transitions overlap, only the transition with the largest $g_{mid}(m_i)$ is retained.
```

Three state probabilities need to be calculated for each frame in the candidate segment. Given a frame in the segment, we extend it forward and backward by

8 frames respectively to form a segment with a length of 17 as the input of the gradual transition detection network. When it is the first or last frame of the video, it needs to be looped multiple times. After obtaining three probability values of each frame, the probability signal $f_M(t)$ is defined according to the following equation, where M ∈(start, mid, end) and s_t represents the segment centered on frame t.

$$f_M(t) = Pr(M \mid s_t) \tag{3}$$

Figure 2 gives an example. Although the original probability signal can indicate the occurrence of the gradual transition, it is not very smooth. We perform window function on $f_M(t)$ to obtain the smooth signal $g_M(t)$. In the next processing, we first determine the transition center by the peak value in $g_{mid}(t)$. Then, a scan is performed within a limited range along the time axis to determine if there are strong peaks in $g_{start}(t)$ and $g_{end}(t)$. If so, the gradual transition boundary can be localized based on these strong peaks. The process of gradual transition detection is described in Algorithm 1.

In the experiment, Th_{mid} is set to 0.5, *separation* is set to 40, $Th_{s,e}$ is set to 0.5. For gradual transitions with a length of 1 or 2 frames, there may not be a maximum point in $g_{mid}(t)$ that meets the requirements in Algorithm 1. Thus, we add steps to detect such transition. Traverse $g_{start}(t)$, if there is a strong peak point s_i that is not included in the found gradual transition, and a strong peak point e_i of $g_{end}(t)$ is found in $[s_i$,$s_i + 1]$, the interval $[s_i$,$e_i]$ is considered as the gradual transition.

4 Experiments

In this part, we will illustrate the experiments on candidate shot boundary detection, abrupt detection and gradual transition detection. All experiments are performed on the ClipShots dataset.

4.1 Evaluation of Candidate Shot Boundary Detection

Evaluation of Different Parameter Values. The parameters λ and γ specify change rate of the difference value between frames and threshold respectively, which control the strictness of the decision-making conditions in step four in Sect. 3.1. We compare the performance of the algorithm with different parameter values, and the results are shown in Table 1. We calculate the ratios of the candidate boundary frames (CBF) to the total frames (TF), the retained abrupts (RA) to the total abrupts (TA), the retained gradual transitions (RG) to the total gradual transitions (TG). It can be seen that as the parameter value increases, more non-boundary frames will be filtered, and more real shot transitions will be lost at the same time. The loss rate of the gradual transition is higher than that of the abrupt, which is in line with the rule that the visual content changes less during the gradual transition.

Evaluation of Different Features. The output of pool10 of SqueezeNet, fc6 of VGG-16 and pool5 of ResNet-50 trained on ImageNet are directly used as the feature representation. Table 2 lists the performance of these models. In addition to the three indicators in Table 1, we also calculate the speed. We adjust the values of λ and γ to make the total number of candidate boundaries obtained on different features close. It can be seen that these three models can achieve better results on shot boundary detection, especially abrupt. SqueezeNet with the fewest parameters is the fastest.

Table 1. Performance of candidate shot boundary detection with different parameter values

Parameter values	CBF/TF (%)	RA/TA (%)	RG/TG (%)
$\lambda = 2$, $\gamma = 0.7$	16.56	99.23	97.91
$\lambda = 2.5$, $\gamma = 0.9$	10.65	98.92	96.6
$\lambda = 3$, $\gamma = 1$	8	98.63	94.77
$\lambda = 3.5$, $\gamma = 1.2$	5.93	98.28	92.29

Table 2. Performance of candidate shot boundary detection with different features

Feature	CBF/TF (%)	RA/TA (%)	RG/TG (%)	Speed (Frames Per Second)
SqueezeNet	16.23	99.23	97.91	1000
VGG-16	16.1	99.16	97.81	129.9
ResNet-50	16.26	99.05	97.70	250

4.2 Evaluation of Abrupt Detection

Training Set. We rebuilt the training set. First, the candidate boundary detection is performed on all videos in the ClipShots training set to obtain a set of video frames. Then, some sampling operations are executed on the video frame set: (1) Sample all video frames whose ground truth is abrupt. (2) Sample all video frames whose ground truth is gradual transition. (3) Randomly sample the video frames whose ground truth is non-boundary, and the number of frames is equal to the sum of the abrupt and gradual frames. In the end, we obtain 116017 abrupt frames, 58623 gradual frames and 174640 non-boundary frames.

Implementation Detail. The size of the input image is 112×112. We use the 3D ResNet-50 pre-trained on the Kinetics dataset published in [21] to initialize the network. SGD is adopted to update the parameters and the momentum is set to 0.9. The batch size is 64 and the initial learning rate is set to 0.001.

Performance. Table 3 shows the results of abrupt detection. We first perform the candidate shot boundary detection on all videos in the ClipShots test set and then perform abrupt detection on the previous output. For comparison, we also add the experimental results of [17–19], which are derived from [19].

Table 3. Performance of abrupt detection on ClipShots

Method	Precision (%)	Recall (%)	F1-measure (%)
DeepSBD [17]	73.1	92.1	81.5
FCN [18]	41.0	9.3	15.1
DSM [19]	77.6	93.4	84.8
Ours (without post-processing)	87.1	90.9	89.0
Ours (with post-processing)	90.8	89.7	90.3

It can be seen that the precision and F1-measure of the proposed model are the highest, improving by at least 10% and 4%, but the recall is lower than DeepSBD and DSM. Compared with DeepSBD and FCN which adopt 8 and 4 convolutional layers, we employ a deeper network with 50 layers to extract features, so the learning and representation capabilities of video content are stronger. Compared with DSM, we adopt the 3D CNN which performs spatio-temporal convolution in all convolutional layers, while in DSM, the input multi-frame is simply regarded as a multi-channel image, which is equivalent to only fusing the temporal information of the video in the first convolutional layer. This is not enough for the spatiotemporal analysis of the input segment. In addition, with post-processing, the precision is increased by 3.7%, but the recall is reduced by 1.2%, and F1-measure is only increased by 1.3%. This shows that post-processing has limited improvement on abrupt detection.

4.3 Evaluation of Gradual Transition Detection

Original Label Translation. Training gradual transition detection network requires three labels: y_{start}, y_{mid} and y_{end}. Due to the extreme imbalance of positive and negative samples (especially y_{start} and y_{end}), and the high similarity of frames near the long-span gradual transition but with different labels, simple 0, 1 labels makes CNN learning unstable. Inspired by [22], we translate y_{start} and y_{end} to force the label of the frames near the gradual transition to be greater than 0. As a result, we can minimize the difference between positive and negative samples while increasing the tolerance for similar training data.

Training Set. The training set is constructed from ClipShots and only_gradual [19]. Sampling four frames from each gradual transition, of which three frames must be the start, mid and end frames, and the last frame is randomly selected. One frame is randomly sampled in the range of 21 frames before and after the gradual transition, and five frames are randomly sampled from the non-gradual frames. In the end, the training set contains 208296 samples, and the ratios of positive and negative for start, mid and end are 1:4.55, 1:2.85, and 1:4.54.

Implementation Detail. We use the 3D ResNet-50 pre-trained on the Kinetics dataset to initialize the body part of the network, and use SGD with momentum of 0.9 to update the parameters. The batch size is 50 and the initial learning rate is set to 0.001.

Performance. Table 4 lists the performance of the gradual transition detection on ClipShots. We perform a complete shot boundary detection process on all videos in the ClipShots test set and the comparison results come from [19].

Table 4. Performance of gradual transition detection on ClipShots

Method	Precision (%)	Recall (%)	F1-measure (%)
DeepSBD [17]	83.7	38.6	52.8
FCN [18]	39.3	5.3	9.3
DSM [19]	84.0	90.4	87.0
Ours	88.1	81.3	84.6

It can be seen that the proposed model performs better than DeepSBD and FCN due to the deeper network. However, even though the network in DSM has only 18 layers, the F1-measure of ours is 2.4% lower than it. The reasons are as follows: (1) The input of the gradual transition detection model is the output of the previous stage where 7.2% of the ground truth has been lost. This directly leads to a low recall. (2) The input length of the model is 17 frames and down-sampling is performed on multiple convolutional layers, while the input length of DSM is 64 frames without any down-sampling operation, making full use of temporal information.

5 Conclusion

In this paper, we propose a shot boundary detection framework based on deep CNN. Three stages are designed to achieve fast and accurate performance, namely candidate shot boundary detection, abrupt detection and gradual transition detection. We introduce a scheme in gradual transition detection, which is to determine the position of the gradual transition by calculating the probability signals of the start, mid and end of the transition. Our method achieves better results on ClipShots dataset. One existing drawback of the proposed method is that there is still a large number of repeated calculations in gradual transition detection. In addition, the mining of difficult negative samples is insufficient. In future work, we will try to improve the network structure and add more negative samples in training to improve the robustness of the model to sudden light change, fast motion and occlusion.

References

1. Zhang, C., Wang, W.: A robust and efficient shot boundary detection approach based on fished criterion. In: 20th ACM International Conference on Multimedia, pp. 701–704 (2012)
2. Lu, Z.-M., Shi, Y.: Fast video shot boundary detection based on SVD and pattern matching. IEEE Trans. Image Process. **22**(12), 5136–5145 (2013)
3. Adjeroh, D.-A., Lee, M.-C., Banda, N., Kandaswamy, U.: Adaptive edge-oriented shot boundary detection. J. Image Video Proc. **2009**, 859371 (2009)
4. Cernekova, Z., Pitas, I., Nikou, C.: Information theory-based shot cut/fade detection and video summarization. IEEE Trans. Circ. Syst. Video Technol. **16**(1), 82–91 (2006)
5. Priya, L., Domnic, S.: Walsh-hadamard transform kernel-based feature vector for shot boundary detection. IEEE Trans. Image Process. **23**(12), 5187–5197 (2014)
6. Apostolidis, E., Mezaris, V.: Fast shot segmentation combining global and local visual descriptors. 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6583–6587. IEEE (2014)
7. Mohanta, P.-P., Saha, S.-K., Chanda, B.: A model-based shot boundary detection technique using frame transition parameters. IEEE Trans. Multimedia **14**(1), 223–233 (2012)
8. Lian, S.: Automatic video temporal segmentation based on multiple features. Soft Comput. **15**(3), 469–482 (2011)
9. Baraldi, L., Grana, C., Cucchiara, R.: Shot and scene detection via hierarchical clustering for re-using broadcast video. In: Azzopardi, G., Petkov, N. (eds.) CAIP 2015. LNCS, vol. 9256, pp. 801–811. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23192-1_67
10. Lankinen, J., Kamarainen, J.-K.: Video shot boundary detection using visual bag-of-words. In: International Conference on Computer Vision Theory and Applications, pp. 788–791 (2013)
11. Thounaojam, D.-M., Thonga, M.K., Singh, K.-M., Roy, S.: A genetic algorithm and fuzzy logic approach for video shot boundary detection. Comput. Intell. Neurosci. **2016**, 1–11 (2016)
12. Yusoff, Y., Christmas, W.-J., Kittler, J.: Video shot cut detection using adaptive thresholding. In: Proceedings of the British Machine Conference, pp. 1–10. BMVA Press (2000)
13. Wu, X., Yuan, P.-C., Liu, C., Huang, J.: Shot boundary detection: an information saliency approach. In: 2008 Congress on Image and Signal Processing, pp. 808–812. IEEE (2010)
14. Xia, D., Deng, X., Zeng, Q.: Shot boundary detection based on difference sequences of mutual information. In: 4th International Conference on Image and Graphics (ICIG 2007), pp. 389–394. IEEE (2007)
15. Tippaya, S., Sitjongsataporn, S., Tan, T., Khan, M.-M., Chamnongthai, K.: Multimodal visual features-based video shot boundary detection. IEEE Access **5**, 12563–12575 (2017)

16. Tong, W., Song, L., Yang, X., Qu, H., Xie, R.: CNN-based shot boundary detection and video annotation. In: 2015 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, pp. 1–5. IEEE (2015)
17. Hassanien, A., Elgharib, M.-A., Selim, A., Hefeeda, M., Matusik, W.: Large-scale, fast and accurate shot boundary detection through spatio-temporal convolutional neural networks. arXiv preprint arXiv: 1705.03281 (2017)
18. Gygli, M.: Ridiculously fast shot boundary detection with fully convolutional neural networks. In: 2018 International Conference on Content-Based Multimedia Indexing, pp. 1–4. IEEE (2018)
19. Tang, S., Feng, L., Kuang, Z., Chen, Y., Zhang, W.: Fast video shot transition localization with deep structured models. In: Jawahar, C.V., Li, H., Mori, G., Schindler, K. (eds.) ACCV 2018. LNCS, vol. 11361, pp. 577–592. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-20887-5_36
20. Nibali, A., He, Z., Morgan, S., Greenwood, G.: Extraction and classification of diving clips from continuous video footage. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 94–104. IEEE (2017)
21. Hara, K., Kataoka, H., Satoh, Y.: Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and imagenet?. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6546–6555. IEEE (2018)
22. Victor, B., He, Z., Morgan, S., Miniutti, D.: Continuous video to simple signals for swimming stroke detection with convolutional neural networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 122–131. IEEE (2017)