Video Highlight Detection via Deep Ranking Modeling

Yifan Jiao[1], Xiaoshan Yang[2], Tianzhu Zhang[2], Shucheng Huang[1(✉)], and Changsheng Xu[2]

[1] School of Computer, Jiangsu University of Science and Technology, Zhenjiang 212003, China
schuang2015@gmail.com

[2] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

Abstract. The video highlight detection task is to localize key elements (moments of user's major or special interest) in a video. Most of existing highlight detection approaches extract features from the video segment as a whole without considering the difference of local features both temporally and spatially. Due to the complexity of video content, this kind of mixed features will impact the final highlight prediction. In temporal extent, not all frames are worth watching because some of them only contain background of the environment without human or other moving objects. In spatial extent, it is similar that not all regions in each frame are highlights especially when there are lots of clutters in the background. To solve the above problem, we propose a novel attention model which can automatically localize the key elements in a video without any extra supervised annotations. Specifically, the proposed attention model produces attention weights of local regions along both the spatial and temporal dimensions of the video segment. The regions of key elements in the video will be strengthened with large weights. Thus more effective feature of the video segment is obtained to predict the highlight score. The proposed attention scheme can be easily integrated into a conventional end-to-end deep ranking model which aims to learn a deep neural network to compute the highlight score of each video segment. Extensive experimental results on the YouTube dataset demonstrate that the proposed approach achieves significant improvement over state-of-the-art methods.

Keywords: Video highlight detection · Attention model
Deep ranking

1 Introduction

With the enormous growth of wearable devices, social media websites (e.g., Flickr, YouTube, Facebook, and Google News) have been rapidly developed in recent years. On YouTube alone, 6000 min of videos are uploaded and 2 million minutes of videos are browsed per minute, which narrows the distance among people and makes them know everything around the world without leaving home.

M. Paul et al. (Eds.): PSIVT 2017, LNCS 10749, pp. 28–39, 2018.
https://doi.org/10.1007/978-3-319-75786-5_3

However, most of the existing videos on the Internet are user-generated. People upload original videos in a variety of time and places for different purposes, which makes most of videos vary in length from a few minutes to a few hours and be full of noise (e.g., severe camera motion, varied illumination conditions, cluttered background) [1,2]. It is a time-consuming and laborious job to browse, edit and index these redundant videos [3–5]. Therefore, highlight detection, which automatically produces the most informative parts of a full-length video [4], has been becoming increasingly important to alleviate this burden.

In recent years, because of its practical value, highlight detection has been extensively studied including two main directions. **(1)** The rule-based approaches [6–13]. This kind of approach generally utilizes heuristic rules to select a collection of frames. Some of them detect highlight by making full use of clear shot boundaries [10–13]. Other methods take advantage of the well defined structure of specific videos in comparison to other egocentric videos [6–9]. A long video can be divided into several components and only a few of them contain certain well defined highlights, such as the score event in soccer games and the hit moment in baseball games. Though the above methods are effective for highlight detection in specific videos (e.g. sports video), they may not generalize well to generic and unstructured videos. **(2)** The ranking-based approaches [4,14,15]. This kind of method treats the highlight detection as scoring each video segment in terms of visual importance and interestingness. The segments with higher scores will be selected as video highlights. Recently, Sun et al. [4] propose a ranking SVM model which outperforms the former approaches. More recently, the landscape of computer vision has been drastically altered and pushed forward through the adoption of a fast, scalable, end-to-end learning framework, the Convolutional Neural Network (CNN), which makes us see a cornucopia of CNN-based models achieving state-of-the-art results in classification, localization, semantic segmentation and action recognition tasks. Based on the Convolutional Neural Network, Yao et al. [15] propose a novel deep ranking model that employs deep learning techniques to learn the relationship between highlight and non-highlight video segments. Here, the relationship of video segments can characterize the relative preferences of all segments within a video and benefit video highlight detection.

However, most of the previous rule-based and ranking-based approaches extract the feature from the video segment as a whole without considering the difference of local features both temporally and spatially. Due to the complexity of the videos, this kind of mixed features will have an impact on the final highlight prediction. In temporal or spatial extent, not all frames or their regions are worth watching because some of them only contain background without human or other moving objects.

To solve the above problem, we propose a novel attention model to automatically localize the key elements in a video without any extra supervised annotations. Specifically, the proposed attention model produces attention weights of local regions along both the spatial and temporal dimensions of the video segment. The regions of key elements in the video will be strengthened with large weights. Thus we can obtain more accurate feature representations of the video segment to effectively predict the highlight score. The proposed attention

scheme can be easily integrated into the conventional deep ranking model. Our method utilizes a two-stream pairwise end-to-end neural network, which aims to learn a function that can denote the highlight score of each video segment for highlight detection. The higher the score, the more highlighted the segment. Segments with higher scores can be selected as video highlights.

Compared with existing methods, the contributions of this paper can be concluded as follows.

1. We propose a novel attention scheme to localize the key elements in a video both spatially and temporally without any extra supervised annotations.
2. We propose an end-to-end highlight detection framework by integrating the attention scheme into deep ranking model as an attention module.
3. Extensive experimental results demonstrate that the proposed attention-based deep ranking model consistently and significantly outperforms existing methods.

The rest of the paper is organized as follows: We present the architecture of our end-to-end deep convolutional neural network in Sect. 2, while Sect. 3 describes the procedure and results of experiments. Finally, conclusion is followed in Sect. 4.

2 The Proposed Method

In this section, we first show the formulation of the highlight detection problem, and then introduce the architecture of our end-to-end deep convolutional neural network including feature module, attention module and ranking module.

2.1 Problem Formulation

Suppose we have a set of pairs Q, in which each pair (p_i, n_i) consists of a highlight video segment p_i and a non-highlight segment n_i. The video segment is comprised of N frames from a raw video. Our goal is to learn a function $f(\cdot)$ which can transform a video segment to the highlight score. This function needs to meet the requirement that the score of highlight segment is higher than the score of the non-highlight segment:

$$f(p_i) > f(n_i), \forall (p_i, n_i) \in Q \tag{1}$$

In this work, the function $f(\cdot)$ is practically a deep neural network which will be illustrated in Sect. 2.2. To achieve the above goal, we learn the function $f(\cdot)$ by minimizing the following loss function:

$$L = \frac{1}{|Q|} \sum_{(p_i, n_i) \in Q} L_Q(p_i, n_i) + \lambda ||\Theta||_F^2 \tag{2}$$

$$L_Q(p_i, n_i) = max(0, 1 - f(p_i) + f(n_i)), (p_i, n_i) \in Q. \tag{3}$$

Here, Θ contains all parameters of the function $f(\cdot)$. The λ is used to control the regularization item. $L_Q(p_i, n_i)$ denotes the contrastive constraint for each pair (p_i, n_i), which is inspired by [17,18]. $|Q|$ is the number of pairs.

2.2 Network Architecture

The deep neural network $f(\cdot)$ introduced in Sect. 2.1 contains three main parts which are feature module, attention module and ranking module as shown in Fig. 1. The network can output the highlight score for any input video segment comprised of N frames. Firstly, the visual feature tensor with size $W \times H \times C$ will be extracted by the feature module for each of the frames in the segment. Then, all the N feature tensors (blue color in Fig. 1) will be input to the attention module which can produce the N attention weight tensors (red color in Fig. 1) which have same size as the feature tensors. By employing weighted aggregation between the visual feature tensor and attention weight tensor, the more effective C-dimension visual feature representation (green color in Fig. 1) is obtained. Finally, the ranking module will output the highlight value. Details of the feature module, the attention module and the ranking module will be illustrated as follows.

Feature Module. This module is comprised of 5 convolution layers and several pooling layers. The exact size of each layer is also shown in Fig. 1. For the j^{th}

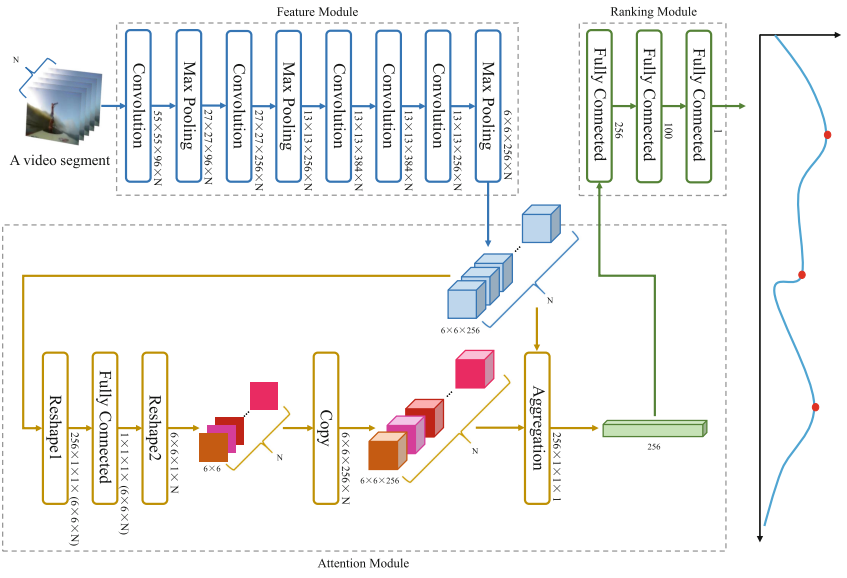

Fig. 1. The flowchart of the proposed end-to-end attention-based deep ranking neural network including three parts: feature module, attention module and ranking module. The input is a raw video segment consists of N frames. The function of attention module is to select the important local regions along spatial and temporal dimensions simultaneously. Then the ranking module predicts the highlight score. By assigning a highlight score to each segment, a highlight curve can be obtained for the full-length video. Next to each layer is the exact size of the output used in our implementation based on Caffe [16]. (Color figure online)

convolution layer, we denote its output as $h_j = s(\boldsymbol{w}_j * h_{j-1} + \boldsymbol{b}_j)$, $j \in \{1, ..., 5\}$. Here, $*$ denotes the convolutional operation, $\boldsymbol{w}_j$ and $\boldsymbol{b}_j$ are the convolutional kernel and bias. $s(\cdot) = max(0, \cdot)$ denotes the non-saturating nonlinearity activation function which is also used as the rectified linear units (ReLU) in [19]. We use $\boldsymbol{a} \in \mathbb{R}^{W \times H \times C \times N}$ to denote the final output of the feature module for all N frames in the input video segment.

Attention Module. This is the core module of our network which is a new and complicated attention scheme we designed. As we all know, given a raw video, not all regions in each frame are useful for highlight detection in spatial extent especially when there are lots of clutters in the background. It is similar in temporal extent that not all frames are worth watching because some frames just contain background of the environment without humans or other moving objects. Considering the aforementioned circumstance, the attention module aims to learn an attention function which can output positive attention weights of local regions along both the spatial and temporal dimensions of the video segment simultaneously.

This module mainly consists of two reshape layers, one fully connected layer, one copy layer and one aggregation layer which will be illustrated as follows.

- **Reshape1**. The output feature of the feature module is denoted as $\boldsymbol{a} \in \mathbb{R}^{W \times H \times C \times N}$ which contains the C feature maps with size $W \times H$ for each of the N frames in the video segment. Since the convolution layer retains the spatial information of the raw input frames and the feature map is always much smaller than the frame image, each element in the feature map corresponds to a specific local region in the original frame. All frame images can be implicitly divided into $W \times H$ regions and each of them corresponds to a specific C-dimensional feature vector in $\boldsymbol{a}$. We can decide whether a local region in the video frames is important for the highlight detection according to the corresponding C-dimensional visual feature.
  To compute the attention weight based on the C-dimensional visual feature, we first need to transform the feature tensor $\boldsymbol{a}$ into a matrix so that the following computing of the attention weight can be easily carried out. Reshape1 layer is used to change the size of the feature tensor $\boldsymbol{a}$ from $W \times H \times C \times N$ to $C \times W \times H \times N$ first and then to a matrix $\boldsymbol{R} \in \mathbb{R}^{C \times (WHN)}$. More explicitly, we also can rewrite the matrix $\boldsymbol{R}$ as the following column vectors:

$$\boldsymbol{R} = [\boldsymbol{r}_1, ..., \boldsymbol{r}_2, ..., \boldsymbol{r}_{WHN}], \boldsymbol{r}_i \in \mathbb{R}^C \tag{4}$$

- **Fully Connected**. This layer in the attention module is designed to compute the attention weights of local regions along both the spatial and temporal dimensions of the video segment simultaneously. We denote the output of the Fully Connected layer as a vector $\alpha \in \mathbb{R}^{HWN}$ which can be computed as follows.

$$\alpha = \sigma(\boldsymbol{w}_\alpha^\top \boldsymbol{R} + b_\alpha) \tag{5}$$

Here, $\boldsymbol{w}_\alpha \in \mathbb{R}^C$ and b_α are the weight matrix and bias. $\boldsymbol{R}$ denotes the output from Reshape1 layer. $\sigma(\cdot)$ is the sigmoid function which is used to control the value of attention weight ranging from 0 to 1.

- **Reshape2**. In the Reshape1 layer, we transform the feature tensor $\boldsymbol{a}$ of the video segment into the matrix $\boldsymbol{R}$ for the convenience of computing attention weight. Here, to match the attention weight vector α with $\boldsymbol{a}$ spatially and temporally, we transform it back into a tensor with size $W \times H \times N$.
- **Copy**. In the previous Reshape2 layer, the attention weight α is transformed into a tensor with size $W \times H \times N$ which is still different from the size of the feature $\boldsymbol{a}$. In order to make them exactly the same, we adopt the Copy layer which transformations the size of α from $W \times H \times N$ to $W \times H \times C \times N$ by making C copies of the weight for each of the WH local regions. This means that, for each of the frames in the video segment, the attention weight tensor is comprised of C identical attention weight matrices of the size $W \times H$.
- **Aggregation**. After obtaining the attention weight α of the video segment, we adopt the Aggregation layer to compute the C-dimensional output feature $\boldsymbol{z}$ of the attention module as

$$\boldsymbol{z}_k = \sum_{l=1}^{N} \sum_{j=1}^{H} \sum_{i=1}^{W} \boldsymbol{a}_{ijkl} \alpha_{ijkl}, \quad k = 1, ..., C \tag{6}$$

The two inner $\sum$ operations are the weighted aggregation spatially and the outer $\sum$ is the weighted aggregation temporally. It is worth noting that we also compute the output of the attention module by replacing $\sum$ in Eq. 6 with max. In the experiment, $\sum$ operation is simply implemented by average pooling while the max operation is implemented by max-pooling.

Through the attention weight α, the more important local regions in the video segment for highlight detection will be paid more attentions in the following ranking module.

Ranking Module. This module mainly consists of three fully connected layers (denoted by F with the number of neurons) which are $\mathrm{F256} - \mathrm{F100} - \mathrm{F1}$. For the k^{th} fully connected layer, we denote the output as $h_k = s(\boldsymbol{w}_k h_{k-1} + \boldsymbol{b}_k)$, $k \in \{1, ..., 3\}$. Here, $\boldsymbol{w}_k$ and $\boldsymbol{b}_k$ are the weight matrix and bias, respectively. For the activation function, the same rectified linear units $s(\cdot) = max(0, \cdot)$ as in the convolution layer of the feature module is adopted. The final output of the ranking module will be taken as the highlight score.

3 Experiments

In this section, we evaluate the performance of the proposed attention-based deep ranking model against several state-of-the-art methods on one public dataset [4].

3.1 Dataset

We evaluate the proposed algorithm on one public dataset. Details of this dataset are illustrated as follows.

YouTube dataset [4]. This dataset contains six normal activities: "gymnastics", "parkour", "skating", "skiing", "surfing" and "dog". Except "skiing" and "surfing" which include near 90 videos, for each domain, there are about 50 videos with various durations. The total time is about 1430 min, which is similar with the state-of-the-art large scale action recognition dataset [20]. This dataset is divided into training and test sets, and each of them covers the half of the videos. A raw video contains several segments and each segment is annotated on a three point ordinal scale: 1-highlight; 0-normal; -1-non-highlight. At the same time, each segment includes approximately 100 frames.

3.2 Baselines

We compare three variants of our method with three state-of-the-art baseline methods:

(1) **LR**, which is short for latent ranking [4]. This is a latent linear ranking SVM model which is trained with the harvested noisy data by introducing latent variables to accommodate variation of highlight selection. They use the EM-like self-paced model selection procedure to train the model effectively.

(2) **DCNN-A**, which stands for deep convolution neural network model [15]. This method uses a convolutional network to extract features and detect highlight by a ranking network. The extractor produces N vectors of a video segment consisting of N frames, each of which is a D-dimensional representation corresponding to one frame. For comparison, we use the average-pooling for the CNN feature representation before feeding into the ranking network with dimension from $D \times N$ to $D \times 1$ similar to the Aggregation layer in the attention module described in Sect. 2.2.

(3) **DCNN-M**, which is short for deep convolution neural network model [15]. The max-pooling operation similar to the Aggregation layer in the attention module described in Sect. 2.2 is applied on the features of a video segment.

(4) **Att-F-A** is our method with a fully connected layer to learn the attention weights and average-pooling operation for the features of a video segment both spatially and temporally in the attention module as described in Sect. 2.2.

(5) **Att-C-M** is our method with a convolution layer which just replaces the fully connected layer in the attention module described in Sect. 2.2 and shown in Fig. 1 to learn the attention weights and max-pooling operation for the features of a video segment both spatially and temporally in the attention module as described in Sect. 2.2.

(6) **Att-F-M** is our method with a fully connected layer to learn the attention weights and max-pooling operation for the features of a video segment both spatially and temporally in the attention module as described in Sect. 2.2.

The **DCNN-A** and **DCNN-M** methods are almost the same except the pooling operation before feeding into the ranking network. **DCNN-A** employs

the average-pooling for the CNN feature extracted from the convolutional network while **DCNN-M** utilizes the max-pooling. The difference between **Att-F-A** and **Att-C-M** or **Att-F-M** is that **Att-C-M** or **Att-F-M** utilizes the max-pooling in the Aggregation layer in the attention module, while **Att-F-A** employs the average-pooling. In order to be more compact, two ways including convolution layer (**Att-C-M**) and fully connected layer (**Att-F-M**) based on the same algorithm and processing are both implemented to learn the attention weights in the attention module.

3.3 Implementation Details

Feature Representation. We initialize the parameters of the feature module in the proposed method according to AlexNet [19], and allow the parameters being fine-tuned. For each frame in an input video segment, the size of the output by the feature module is $6 \times 6 \times 256$. By employing a weighted aggregation between the visual feature tensor and attention weight tensor, the more effective feature representation is obtained for a video.

Evaluation Metrics. The pairwise accuracy Acc_v for the v^{th} video is defined as

$$Acc_v = \frac{\sum\limits_{(p_i,n_i)\in\mathbf{Q}} [f(p_i) > f(n_i)]}{|Q|} \tag{7}$$

Where $[\cdot]$ is the Iverson bracket notation. $f(p_i)$ represents the final output score of a highlight segment, and $f(n_i)$ is the value of a non-highlight one. As described in Sect. 2.1, Q is the set of pairwise constraints for the v^{th} video, and $|Q|$ is the number of pairs. The accuracy Acc_v is a normalized value ranging from 0 to 1. The higher the value, the more accurate the detection. We calculate the average precision of highlight detection for comparison with baseline methods.

Training Process. Since the data annotation on YouTube dataset [4] is not completely accurate in some videos, which means the segment annotated as highlight may be not correct in fact, it is a weak supervision dataset. We select a set of latent best highlighted segments iteratively using the EM-like approach as described in [4] to train the baseline methods.

3.4 Performance Comparison

Youtube dataset. Figure 2 summarizes the overall highlight detection results for different methods on YouTube dataset [4], and shows that our proposed approach significantly outperforms the state-of-the-art methods in all six domains on the public dataset based on the same evaluation metrics. In particular, the precision of "parkour" and "surfing" can achieve 0.75 and 0.78, which makes the considerable improvement over the compared methods, as they do not consider the region information in a frame spatially and the importance of each frame temporally. Instead, our approach outputs the attention weights to find the important regions and frames, which helps us obtain more accurate feature

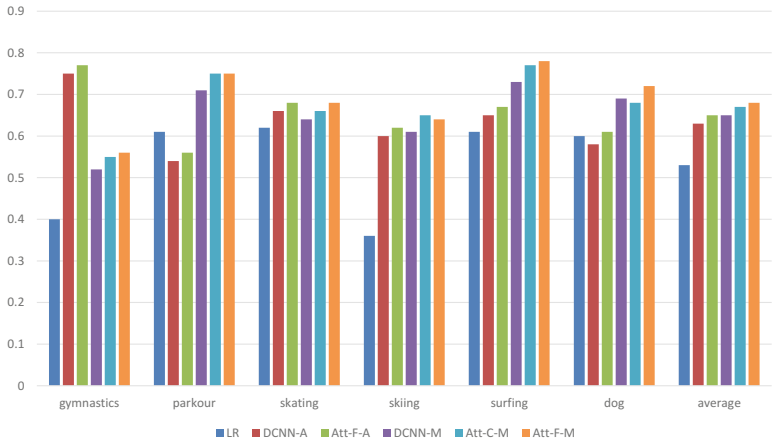

Fig. 2. Performance comparison between our attention-based ranking model and other baseline approaches. Our method shows better performance.

representation of videos for effective highlight detection. On average, the mean average precision of our attention-based model with **Att-F-M** can achieve 0.68, which makes the improvement over **LR**, **DCNN-A** and **DCNN-M** by 15%, 5% and 3% respectively.

In general, the result of max-pooling (**DCNN-M**, **Att-C-M**, **Att-F-M**) is better than that of average-pooling (**DCNN-A**, **Att-F-A**) in Fig. 2. The main reason may be that, compared with the average-pooling, the max-pooling is more likely to retain the feature of the most important local regions.

3.5 Visualization

Figure 3 shows several raw frames sampled in a segment and the corresponding heat maps of attention weight from YouTube dataset. The lighter the color, the bigger the weight. Therefore, the white regions in the heat map reflect that the corresponding regions in video frame contribute more for highlight detection than other regions. For instance, in the first ten frames selected from a segment of the parkour domain in Fig. 3, the boy is ready to run and another man is looking at him in the first frame, so the two white square regions in first heat map approximately describe the location of them, which means these regions are more important for highlight detection. When he runs approach to the tree, the white regions are moving as well in the second frame. He is jumping in the third frame, the white regions are also changing according to the location of the boy. It is similar for the rest seven frames that the white regions corresponding to the boy's movement are important for highlight detection spatially. The lightness of the ten heat maps are nearly the same, which means they are all important frames in this segment temporally. The heat maps of the first and second frames in the third group are lighter than the rest, which means they are more important than the other frames in the segment for highlight detection.

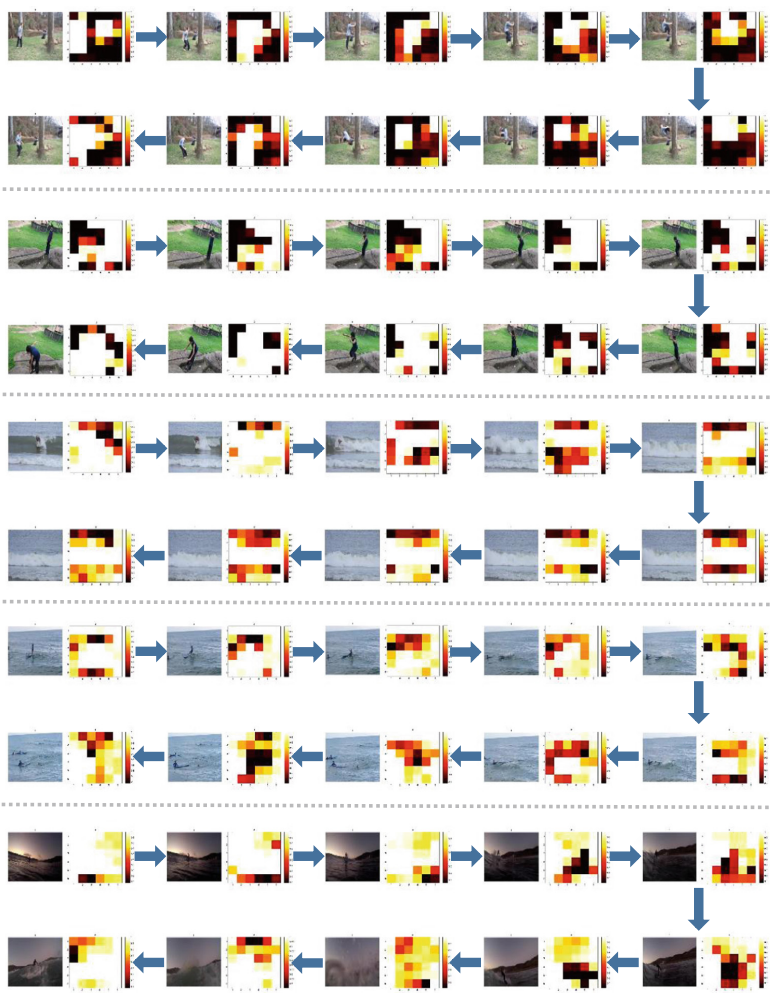

Fig. 3. Attention scores of each region of a frame in different video domains in [4]. We sample 10 frames from each raw video segment, and draw the heat map of attention weight in detail. The lighter the color, the bigger the weight. In particular, the regions that white color corresponds to are the most significant ones for highlight detection.

4 Conclusion

In this paper, we propose a novel attention-based deep ranking model to learn a function to output attention weights of local regions in the video segment along spatial and temporal dimensions simultaneously, which helps us obtain the more effective feature representation of interesting and significant components in videos for highlight detection. Extensive experiments show that our method performs better than state-of-the-art approaches on one public dataset.

In the future, we would like to develop a highlight-driven video summarization system based on our proposed attention-based model. We will also explore more comprehensive attention scheme to incorporate other useful information.

Acknowledgement. This work is supported in part by the National Natural Science Foundation of China under Grant 61432019, Grant 61572498, Grant 61532009, and Grant 61772244, the Key Research Program of Frontier Sciences, CAS, Grant NO. QYZDJ-SSW-JSC039, the Beijing Natural Science Foundation 4172062, and Postgraduate Research & Practice Innovation Program of Jiangsu Province, Grant NO. SJCX17_0599.

References

1. Liu, S., Wang, C.H., Qian, R.H., Yu, H., Bao, R.: Surveillance video parsing with single frame supervision. arXiv preprint arXiv:1611.09587 (2016)
2. Liu, S., Liang, X.D., Liu, L.Q., Shen, X.H., Yang, J.C., Xu, C.S., Lin, L., Cao, X.C., Yan, S.C.: Matching-CNN meets KNN: quasi-parametric human parsing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1419–1427 (2015)
3. Zhang, T.Z., Liu, S., Ahuja, N., Yang, M.H., Ghanem, B.: Robust visual tracking via consistent low-rank sparse learning. Int. J. Comput. Vis. **111**(2), 171–190 (2015)
4. Sun, M., Farhadi, A., Seitz, S.: Ranking domain-specific highlights by analyzing edited videos. In: ECCV (2014)
5. Liu, S., Feng, J.S., Domokos, C., Xu, H., Huang, J.S., Hu, Z.Z., Yan, S.C.: Fashion parsing with weak color-category labels. IEEE Trans. Multimed. **16**(1), 253–265 (2014)
6. Rui, Y., Gupta, A., Acero, A.: Automatically extracting highlights for TV baseball programs. In: Proceedings of the 8th ACM International Conference on Multimedia 2000, Los Angeles, CA, USA, 30 October–3 November 2000, pp. 105–115 (2000)
7. Nepal, S., Srinivasan, U., Graham, J.R.: Automatic detection of goal segments in basketball videos. In: Proceedings of the 9th ACM International Conference on Multimedia 2001, Ottawa, Ontario, Canada, 30 September–5 October 2001, pp. 261–269 (2001)
8. Otsuka, I., Nakane, K., Divakaran, A., Hatanaka, K., Ogawa, M.: A highlight scene detection and video summarization system using audio feature for a personal video recorder. IEEE Trans. Consum. Electron. **51**(1), 112–116 (2005)
9. Tong, X.F., Liu, Q.S., Zhang, Y.F., Lu, H.Q.: Highlight ranking for sports video browsing. In: Proceedings of the 13th ACM International Conference on Multimedia, Singapore, 6–11 November 2005, pp. 519–522 (2005)
10. Ngo, C., Ma, Y.F., Zhang, H.J.: Video summarization and scene detection by graph modeling. IEEE Trans. Circuits Syst. Video Technol. **15**(2), 296–305 (2005)
11. Mundur, P., Rao, Y., Yesha, Y.: Keyframe-based video summarization using delaunay clustering. Int. J. Dig. Libr. **6**(2), 219–232 (2006)
12. Borth, D., Ulges, A., Schulze, C., Thomas, M.B.: Keyframe extraction for video tagging & summarization. In: Informatiktage 2008. Fachwissenschaftlicher Informatik-Kongress, 14–15 März 2008, B-IT Bonn-Aachen International Center for Information Technology in Bonn, pp. 45–48 (2008)

13. Qu, Z., Lin, L.D., Gao, T.F., Wang, Y.K.: An improved keyframe extraction method based on HSV colour space. JSW **8**(7), 1751–1758 (2013)
14. Lin, Y.L., Vlad, I.M., Winston, H.H.: Summarizing while recording: context-based highlight detection for egocentric videos. In: 2015 IEEE International Conference on Computer Vision Workshop, ICCV Workshops 2015, Santiago, Chile, 7–13 December 2015, pp. 443–451 (2015)
15. Yao, T., Mei, T., Rui, Y.: Highlight detection with pairwise deep ranking for first-person video summarization, pp. 982–990 (2016)
16. Jia, Y.Q., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Ross, B.G., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. CoRR, abs/1408.5093 (2014)
17. Yang, X.S., Zhang, T.Z., Xu, C.S., Yan, S.C., Hossain, M.S., Ghoneim, A.: Deep relative attributes. IEEE Trans. Multimed. **18**(9), 1832–1842 (2016)
18. Gao, J.Y., Zhang, T.Z., Yang, X.S., Xu, C.S.: Deep relative tracking. IEEE Trans. Image Process. **26**(4), 1845–1858 (2017)
19. Krizhevsky, A., Sutskever, I., Geoffrey, E.H.: ImageNet classification with deep convolutional neural networks. In: 26th Annual Conference on Neural Information Processing Systems, pp. 1106–1114 (2012)
20. Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild. volume abs/1212.0402 (2012)