Discriminative and efficient non-local attention network for league of legends highlight detection

Qian Wan 1 · Aruna Wang 1 · Guoshuai Zhang 1 · Le Liu 2 · Jiaji Wu 1

Received: 9 December 2021 / Accepted: 22 April 2022 / Published online: 19 May 2022
© The Author(s) 2022

Abstract
With the growing popularity of eSports, video highlight detection, which encapsulates the most informative parts in a few seconds, has become a critical part of live competition. However, learning the spatial–temporal dependency efficiently and discriminatively in video highlight detection for league of legends (LoL) is a critical problem. In this study, to address these existing problems, we propose a novel discriminative and efficient non-local attention network (DENAN) for LoL highlight detection. In particular, both spatial and temporal dependencies are learned using an end-to-end lightweight trainable framework. An auxiliary triplet loss is used in discriminative training to learn robust LoL video feature representations and improve DENAN’s performance. Our experimental results on the NLACS and LMS datasets show the effectiveness of our method in terms of performance and computation cost.

Keywords Highlight detection · AI · eSports · Non-local attention network

Introduction
Recently, eSports has become increasingly popular among players and viewers. The league of legends (LoL), one of the most famous online eSports, has attracted numerous players and fans around the world. According to a Statista report, the LoL World Championships peaked at approximately 46 million concurrent viewers in 2020. Highlight replay is an important part of live streaming because it shows the most amazing fighting fragments during the live streaming. However, current LoL highlight replay largely manual. Therefore, it is critical to implement automatic and efficient highlight detection in LoL.

As an attempt to automate highlight generation, video highlight detection has attracted the attention of both academics and industry. The goal of highlight detection is to generate a short video clip from the candidate video that captures a user’s primary attention or interest [1]. Existing video highlight detection methods could mainly be divided into two categories: structure-driven [2, 3] and keyframe-based methods [4–6]. Researchers using structure-driven methods have focused on a well-defined data structure in the detected video, such as audience cheering or chatting, score changes, or other special events. The main goal of keyframe-based methods is to optimize the feature representation from the frame level to the clip level. Key-frames are extracted using the histogram of oriented gradient (HOG), scale-invariant feature transform (SIFT), and cluster algorithms, and then video browsing is performed near the selected keyframes, smoothing out video highlights. With the recent advancements in convolutional neural network (CNN), CNN-based methods have been extensively used in video highlight detection [4, 6–8].

To learn a robust feature representation for video highlight detection, some studies have considered the temporal dependency between video frames. This study [6] used long short-term memory (LSTM) [9] to model the variable-range
LoL videos are parsed into a sequence of frames and then fed into DENAN. Each frame will be classified to 1 or 0 corresponding to highlight or not. Finally, several classified consecutive frames with highlights are combined into the LoL video highlights.

Attention mechanism for an efficient and discriminative end-to-end trainable LoL video highlight detection task. Based on another study [13], we propose a discriminative and efficient non-local attention network (DENAN), as shown in Fig. 1, which incorporates the non-local attention mechanism into ShuffleNetV2 [14], a light-weight CNN for classification. The non-local attention module refines the LoL video sequence representations by generating mean-weighted attention to the features of different spatial and temporal locations in the sequences. DENAN explores the spatial–temporal diversity of LoL video sequence, and discriminatively and efficiently learns the sequence representation. The main contributions of this study are as follows:

1. In this study, we propose an end-to-end framework for learning both spatial and temporal dependencies in a discriminative and efficient non-local attention network for LoL video highlight detection.
2. We significantly reduce the computation cost for LoL video highlight detection task while significantly improving performance.
3. Experimental results on the NALCS and LMS datasets show that our proposed method outperforms existing methods in terms of efficiency and accuracy.

Related work

In this section, we focus on recent methods related to our work, which includes video highlight detection, eSports video highlight detection, and attention-based video highlight detection.
Video highlight detection

Research on video highlight detection is mainly performed along two directions: (a) keyframe-based methods and (b) structure-driven methods. Keyframe-based methods use a subset of representative keyframes from the original video to generate highlights. Most early video highlight detection methods focus on extracting keyframes independently and using them as a classification task. Borth et al. proposed a keyframe extraction approach in which the video is first segmented into shots using shot boundary detection, and then obtained keyframes using the k-means algorithm [15]. Lin et al. used a context-specific highlight support vector machines (SVM) model to summarize video sequences without watching the entire video by predicting the contextual information of each video segment [16]. Even though these methods achieved remarkable performance, they only extract low-level features and ignore the temporal dependency, which describes the relationship between highlight and non-highlight frames.

Unlike keyframe methods, structure-driven methods exploit a well-defined data structure in the detected video, such as audience cheering and chatting, score changes, or other special events. Therefore, structure-based methods are suitable for sports video highlight detection, and they have attracted the attention of many researchers [6, 17, 17]. Zhao et al. proposed a highlight detection model based on audio energy and motion activity. Hsieh et al. proposed a more flexible solution for finding important and meaningful events in sports games by analyzing the messages shared between users on microblog services [18]. While sports video highlight detection has improved, these methods rely on audio, textual, and psychological data, which are not always easy to obtain.

eSports video highlight detection

With the recent rapid development of eSports, video highlight detection has attracted the attention of both industry and academia. Fu et al. proposed a CNN-LSTM model for LoL that combine visual features and real-world audience discourse. Song et al. proposed a cascaded prediction approach for learning convolution filters of visual effects for detecting video highlights in Heroes of the Storm, LoL, and Dota2 [19]. Wang et al. [20] proposed a multi-stream framework to fuse spatial, temporal information, and audio features extracted from Honor of King videos.

Recent eSports video highlight detection attempted to address these problems from a cross-modal perspective. However, for real-world applications, cross-modal information, such as audience chat or audio signal, is either difficult to obtain or requires additional computation cost. Therefore, we attempt to consider eSports video highlight detection using only visual features.

Attention-based video highlight detection

The objective of attention-based video highlight detection is to find what the user is most paying attention to, which is highly correlated with highlights. Ma et al. presented a generic framework in which computational attention models based on the modeling of viewer’s attention were used [21]. Ma et al. presented a generic framework for a user attention model, which estimates the how much attention viewers may pay to video content. Ejaz et al. proposed an efficient visual attention model based on the key frame extraction method and reduced the computational cost using the temporal gradient based on dynamic visual saliency detection [22]. Additionally, some studies integrated spatial–temporal clues into attention-based video highlight detection, with the goal of determining when and where users are most interested. A novel 3D dimensional attention model was proposed, which can automatically localize the key elements in a video without any extra supervised annotations [23].

A self-attention mechanism was proposed to model the long-range dependency in machine translation [24]. Inspired by self-attention mechanism [24] and the non-local means algorithm [25], Wang et al. proposed the non-local attention [13], which computes the response at one position as a weighted sum of the features at all positions, capturing long-range spatial–temporal dependency for video representation. Our work is similar to [23], but [23] computes coarse-grained spatial temporal attention in which the attention matrix only models the relationship between different channels and shares along spatial and temporal dimensions. Conversely, DENAN can capture a fine-grained relationship at the pixel level, capturing the relationship at all positions along spatial and temporal dimensions.

Methods

In this section, we first describe the overview of our proposed DENAN. Then, each sub-module of DENAN will be described in detail.

Network architecture for video highlight detection of LoL

Figure 2 shows an overview of our proposed network, which consists of three parts: (1) a LoL video encoder, which is used to convert LoL video sequences into deep feature representations. ShuffleNetV2 [14] is used as the encoder, (2) non-local attention module, which is used to capture both spatial–temporal and long-range dependencies in LoL video
Fig. 2 The proposed DENAN architecture. (1) Given a sequence of LoL video frames as input, a video encoder transforms them into frame-level feature maps. (2) The resized feature maps are sent to non-local attention module, which captures both spatial and temporal long-range dependencies. (3) Finally, feature maps are fed into spatial–temporal aggregation, dropout, and fully connected (FC) layers.

sequences, and (3) discriminative loss function for training. Triplet loss (TL) [26] and cross entropy (CE) loss are used to optimize the entire framework.

Video encoder for LoL

We adopt an efficient video encoder with low computation costs to meet the requirement for a real-time video highlight detection system. Inspired by the recent studies [14, 27], which designed an efficient and light-weight 2D universal CNN, we adopt ShuffleNetV2 [14] (Fig. 3) as a frame-level video encoder for LoL.

The number of feature channels in light-weight network is limited by the computing resources available. Compared with ShuffleNetV1 [27], the main improvement of ShuffleNetV2 [14] is channel split operation. Particularly, the total number of channels of the input feature map is divided into one branch with $C'$ channels and the other with $(C - C')$ channels ($C$ is the total number of channels). After a three-layer convolution operation for each branch, the features of the two branches are combined. Then, a channel shuffle operation is applied to enable two branches of information to interact.

Given a set of video sequences $\{X_i\}_{i=1}^N$, where $X_i$ denotes one LoL video sequence and $N$ represents the total number of video sequences. Each sequence contains $T$ frames, where $T$ is the video sequence length. Frame-by-frame video images for one sequence are input to the video encoder to obtain a set of frame-level feature sequences $\chi_i = \{x_t^i\}_{t=1}^T$, where $x_t^i$ represent the feature map for rth image in $i$th sequence. Then, the output from the video encoder is fed into the non-local attention module to further capture the spatial–temporal features of LoL video.

Spatial–temporal feature extraction using non-local attention module

The input of non-local attention module are the frame-level feature sequences $\chi_i = \{x_t^i\}_{t=1}^T$. Following the non-local form proposed in [13, 25], the non-local operation could be defined as follows:

![Diagram of video encoder for LoL](image-url)
\[ \mathcal{F}_{i,j} = \frac{1}{N(\chi_i)} \sum_{vk} s(\chi_{i,j}, \chi_{i,k}) g(\chi_{i,k}), \]  
\[ \tilde{\mathcal{F}}_{i,j} = \frac{1}{N} \sum_{vk} \theta(\chi_{i,j})^T \phi(\chi_{i,k}) g(\chi_{i,k}), \]  
where \( \chi_i \) represent the extracted feature sequences, \( j \) represent the computed position index of the response, \( k \) represent the possible positions of all input feature sequences in both spatial and temporal dimensions, \( s(\chi_{i,j}, \chi_{i,k}) \) denotes the relationship between position \( j \) and position \( k \) of the input feature sequences, \( g(\chi_{i,k}) \) computes the feature representation of input \( \chi_i \) at position \( k \), and \( N(\chi_i) \) is the normalization factor.

We design a non-local attention module based on the non-local operation [13] to capture spatial and temporal pixel dependencies in LoL video (Fig. 4). The definition of non-local operation is similar to Eq. (1), which captures the spatial and temporal long-range pixel relationship. Then, the definition of non-local attention is defined as follows:

\[ \tilde{\mathcal{F}}_{i,j} = \frac{1}{N} \sum_{vk} \theta(\chi_{i,j})^T \phi(\chi_{i,k}) g(\chi_{i,k}), \]  
where \( \tilde{\mathcal{F}}_{i,j} \) represent a specific position element of \( \tilde{\mathcal{F}}_i \) on position \( j \), \( k \) are all possible positions of \( \chi_i \) to be computed. Here, we project \( \chi_{i,j} \) and \( \chi_{i,k} \) to embedding space using a linear transformation. Therefore, \( \theta(\chi_{i,j}) = W_\theta \chi_{i,j} \), \( \phi(\chi_{i,k}) = W_\phi \chi_{i,k} \), and \( g(\chi_{i,j}) = W_g \chi_{i,j} \), where \( W_\theta \), \( W_\phi \), and \( W_g \) are weights to be learned. Equation (2) is similar to the self-attention mechanism proposed in [24].

To project \( \tilde{\mathcal{F}}_i \) into the original space, the non-local operation is wrapped into non-local attention module, as shown in Fig. 4, which is defined as follows:

\[ Z_i = W_z \tilde{\mathcal{F}}_i + \chi_i, \]  
where \( W_z \) represent a linear projection matrix, which is implemented using a \( 1 \times 1 \times 1 \) convolution, and \( Z_i \) represent the final output of non-local attention module. A video highlight is a duration of continuous frames that are not time-independent. However, determining whether a particular frame is a highlight must consider the long-range pixel dependency in the spatial dimension. Therefore, we should consider the relationship between LoL videos from both spatial and temporal dimensions.

To further refine the spatial–temporal information of the learned features, we use global average pooling (GAP) to aggregate information in spatial and temporal dimensions as follows:

\[ z_i^c = \frac{1}{T \times H \times W} \sum_{t=1}^{T} \sum_{h=1}^{H} \sum_{w=1}^{W} z_t^c(h, w, c), \]  
where \( \bar{z}_i = [\bar{z}_1, \ldots, \bar{z}_c] \) is the feature representation vector of \( \chi_i \), \( c \) denotes the position index of \( z_i^c \) in \( \bar{z}_i \), and \( c = 1024 \) in our proposed DENAN.

### Loss function for training DENAN

To extract discriminative and robust features from DENAN, we combined CE loss with TL [26] to train our framework for LoL video highlight detection. The CE loss is defined as follows:

\[ L_{CE} = -\frac{1}{N} \sum_{i=1}^{N} y_i \log f(\bar{z}_i). \]  
where \( f(\bar{z}_i) \) denotes softmax, \( W_y \) represent the weight, the output \( f(\bar{z}_i) \) denotes the probability of whether the input frame is a highlight, \( N \) is the number of LoL video sequences in a mini-batch, and \( y_i \) are all classes, including highlight and non-highlight.

Meanwhile, the discriminativeness of the features extracted from DENAN was improved using TL [26]. The batch-hard TL is defined as follows:

\[ \mathcal{T}(\bar{z}_i, \bar{z}_j) = \exp(W_y^T \bar{z}_i) / \sum_j \exp(W_y^T \bar{z}_j), \]  
where \( W_y \) is the weight.
where $\bar{z}_a$, $\bar{z}_p$, $\bar{z}_n$ are features extracted from the anchor, positive and negative samples, respectively, and $\lambda$ is the margin hyper-parameter to control the differences between intra- and inter-distance. Here, positive and negative samples refer to the LoL video sequence with the same or different class from the anchor.

In summary, the loss function for DENAN training is a combination of TL and CE losses, which is defined as follows:

$$L_{\text{total}} = L_{\text{CE}} + \lambda L_{\text{TL}},$$

where $\lambda$ controls the balance of TL and CE loss, $\lambda$ is 1 in this work.

### Experiment

#### Datasets and evaluation metrics

We trained and evaluated our proposed DENAN on two datasets: NALCS [28] and LMS [28]. NALCS contained 218 videos of LoL in the 2017 Spring, with 128 videos divided into a training set, 40 videos divided into a validation set, and 50 videos divided into a test set. The average length of each video is between 30 and 50 minutes, with both highlight and non-highlight frames. The data labeling process has been described in detail [28].

The LMS dataset contained 103 videos of LoL, including 57 training videos, 18 validation videos, and 28 testing videos. In our experiment, training and validation sets of these two datasets were used for training, while test sets were used for testing.

Based on the commonly used metrics in video summarization tasks [6, 8, 29], we use precision (P), recall (R), and F1-score (F1) as evaluation metrics in our experiments to evaluate the performance of DENAN. Let TP denotes positive frames that are correctly predicted, FP denotes positive frames with an incorrect prediction, FN denotes negative frames with an incorrect prediction. Then, the P, R, and F1 are calculated as follows:

$$P = \frac{TP}{TP + FP},$$

$$R = \frac{TP}{TP + FN},$$

$$F1 = \frac{2PR}{P + R}.$$
Table 1 Ablation study using the NALCS and LMS datasets with each DENAN component

| Component | NALCS | LMS | Num params (M) | FLOPs (M) |
|-----------|-------|-----|----------------|-----------|
|           | P     | R   | Fl  | P    | R   | Fl  |               |           |
| × ×       | 0.76  | 0.71| 0.74| 0.69 | 0.78| 0.74| 1.25          | 143.88    |
| × √       | 0.77  | 0.72| 0.75| 0.71 | 0.79| 0.75| 1.25          | 143.88    |
| √ ×       | 0.80  | 0.69| 0.75| 0.76 | 0.74| 0.75| 3.35          | 246.77    |
| √√        | 0.77  | 0.74| 0.76| 0.73 | 0.78| 0.76| 3.35          | 246.77    |

Best in bold
“×” denotes an unused component, while “√” denotes a used component

Table 2 shows the experimental results on NALCS and LMS datasets for an ablation study based on the length of input LoL video sequences. For a fair comparison, we use the model trained with a video sequence length of 16 and evaluate the performance of different lengths (4, 8, and 16). Compared with \( T = 4 \), when \( T = 16 \), P, R, and Fl increased by 0.11, 0.04, and 0.08, respectively, on the NALCS dataset, whereas on the LMS dataset, P, R, and Fl increased by 0.08, 0.08, and 0.08, respectively. This is because our proposed DENAN captures the spatial–temporal dependency between different frames. In theory, as the length of the video sequence grows, the performance of the model should increase. However, the length is limited by the GPU memory, and we can only set a maximum length of 16.

Comparison with state-of-the-art methods

As shown in Table 3, our approach significantly outperforms the state-of-the-art methods based on the evaluation metrics with P, R, Fl, FLOPs, and Num Params on NALCS and LMS datasets.
### Table 2
Ablation study on both NALCS and LMS datasets with different LoL video sequence length

| Length | NALCS | | | LMS | | |
|--------|-------|---|---|---|---|---|
|        | P     | R | F1 | P | R | F1 |
| 4      | 0.66  | 0.70 | 0.68 | 0.65 | 0.70 | 0.68 |
| 8      | 0.73  | 0.67 | 0.70 | 0.66 | 0.73 | 0.70 |
| 16     | 0.77  | 0.74 | 0.76 | 0.73 | 0.78 | 0.76 |

Best in bold

### Table 3
Comparison with state-of-the-art methods on both NALCS and LMS datasets

| Method      | NALCS | | | LMS | | | Num params (M) | FLOPs (M) |
|-------------|-------|---|---|---|---|---|----------------|------------|
|             | P | R | F1 | P | R | F1 |                      |            |
| DR-DSN [8]  | 0.62 | 0.80 | 0.70 | 0.57 | 0.84 | 0.68 | 2.62 | 41.97 |
| Iv-LSTM [28] | 0.79 | 0.70 | 0.75 | 0.72 | 0.68 | 0.70 | 21.35 | 3663.31 |
| Ours        | 0.77 | 0.74 | 0.76 | 0.73 | 0.78 | 0.76 | 3.35 | 246.77 |

Best in bold

---

**Fig. 6** Visualization of LoL video highlight detection made by Tencent and DENAN. Green block denotes highlight frame, and red block denotes non-highlight frame (colour figure online)

In Table 3, P, R, and F1 of the proposed DENAN are 0.77, 0.74, and 0.76, respectively, on the NALCS dataset and 0.73, 0.78, and 0.76, respectively, on the LMS dataset. In particular, the Num Params and FLOPs of the proposed DENAN are 3.35 and 246.77 M, respectively. DR-DSN [8] develops video summarization as a sequential decision-making process and a deep summarization network to indicate the likelihood of a frame being selected to summarize the video. According to the probability distributions, selected frames are used as video highlights. Our method is 0.06 higher for F1 on the NALCS dataset, and 0.08 higher on the LMS dataset. In terms of Num Params and FLOPs, DENAN outperformed DR-DSN. However, DR-DSN first extracts frame-level features using GoogLeNet [31], and the extracted features are
always local. Therefore, our model still has a computational complexity advantage. Iv-LSTM [28] is proposed for video highlight prediction in LoL based on joint visual features and textual analysis of the audience commentary. It achieves 0.79, 0.7, and 0.75, and 0.72, 0.68, and 0.7 on the NALCS and LMS datasets, respectively. Our proposed method outperforms Iv-LSTM by −0.02, 0.04, and 0.01, and 0.01, 0.1, and 0.06 on the NALCS and LMS datasets, respectively. DENAN outperforms Iv-LSTM with only visual features, which demonstrates the effectiveness of DENAN for LoL video highlight detection.

Visualization of DENAN Performance

We compare the results of LoL video highlight made by Tencent and our proposed DENAN. Figure 6a, c show the LoL video highlight replay made by Tencent. Figure 6b, d show the LoL video highlight detection made by our proposed DENAN. Highlight frames and non-highlight frames are marked with green and red blocks, respectively. Most highlight frames are detected correctly, and our proposed DENAN outperforms state-of-the-art methods, including Tencent’s current methods. Our proposed DENAN not only achieves accurate video highlight detection (higher P, R, and FI), but it also has lower FLOPs and Num Params, enabling automatic and real-time video highlight detection.

Figure 7 shows the visualization of our proposed method on LoL videos from the NALCS and LMS datasets. We denote video highlight and non-highlight frames as 1 and −1. The upper part of Fig. 7a, b shows a comparison of ground truth (label) and prediction (predict), where the turquoise line represents the ground truth of each frame in the video, and the light-pink line represents the prediction of our proposed DENAN. The lower part shows the overlap between ground truth and prediction, with green indicating correct prediction and red indicating the opposite. As shown in Fig. 7, most positive labels are predicted correctly, which demonstrates the effectiveness of our method.

Conclusion

In this study, we propose a DENAN for LoL video highlight detection with low computation cost, in which light-weight ShuffleNetV2 video encoder is used to extract frame-level features from LoL video sequences, and non-local attention is used to capture spatial-temporal long-range dependencies. Training with CE loss and TL improves the performance of DENAN. Experimental results on the NALCS and LMS datasets demonstrate the validity of the proposed method.

Acknowledgements

This work was supported by the Key Research and Development Program of Shaanxi Province-Key Industry Innovation Chain (Group)-Industrial Field under No. 2019ZDLGY10-06.

Declarations

Conflict of interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

Open Access

This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Xiong B, Kalantidis Y, Ghabadiaram D, Grauman K (2019) Less is more: learning highlight detection from video duration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1258–1267
2. Nepal S, Srinivasan U, Reynolds G (2001) Automatic detection of “goal” segments in basketball videos. In: Proceedings of the ninth ACM international conference on multimedia, pp 261–269
3. Rui Y, Gupta A, Acero A (2000) Automatically extracting highlights for TV baseball programs. In: Proceedings of the eighth ACM international conference on multimedia, pp 105–115
4. Ma Y-F, Hua X-S, Lie Lu, Zhang H-J (2005) A generic framework of user attention model and its application in video summarization. IEEE Trans Multimed 7(5):907–919, https://doi.org/10.1109/TMM.2005.854410

 Springer
