Words speak for Actions: Using Text to find Video Highlights

Sukanya Kudi and Anoop M. Namboodiri
Center for Visual Information Technology,
KCIS, IIIT-Hyderabad
Hyderabad, India
kudi.sukanya@research.iiit.ac.in, anoop@iiit.ac.in

***Abstract*—Video highlights are a selection of the most interesting parts of a video. The problem of highlight detection has been explored for video domains like egocentric, sports, movies, and surveillance videos. Existing methods are limited to finding visually important parts of the video but does not necessarily learn semantics. Moreover, the available benchmark datasets contain audio muted, single activity, short videos, which lack any context apart from a few keyframes that can be used to understand them. In this work, we explore highlight detection in the TV series domain, which features complex interactions with the surroundings. The existing methods would fare poorly in capturing the video semantics in such videos. To incorporate the importance of dialogues/audio, we propose using the descriptions of shots of the video as cues to learning visual importance. Note that while the audio information is used to determine visual importance during training, the highlight detection still works using only the visual information from videos. We use publicly available text ranking algorithms to rank the descriptions. The ranking scores are used to train a visual pairwise shot ranking model (VPSR) to find the highlights of the video. The results are reported on TV series videos of the VideoSet dataset and a season of *Buffy the Vampire Slayer* TV series.**

***Keywords*-Video Highlights, Pairwise Ranking, TextRank.**

I. Introduction

A Cisco white paper [1] released in 2016 suggested that 70% of the Internet traffic in 2015 was contributed by video content. By 2020, video traffic is expected to occupy 82% of the consumer Internet traffic. Video data arises from a variety of domains, including egocentric, sports, surveillance, news, television and movies. These domains have characteristic features and have been used in various computer vision tasks. The advent of deep learning and powerful computational resources, has made it possible to process a large amount of video data in a reasonable amount of time. Researchers have worked on problems that range from low level tasks such as computing optical flow, object tracking and localization to high level tasks such as action recognition, retrieval, video classification, video summarization and character identification. With such high amount of data available, video understanding problem in computer vision has naturally drawn attention.

We convey our thoughts in the form of language, which is a key to impart semantics. Several attempts have been made to use text in conjunction with videos such as event detection[1] in sports videos, image/video retrieval using the text [2], video description generation [3]. The aim of such work is to learn a semantic space in which the similarity between entities of different modalities is captured.

Text description: Megan and Elaine talk about Elaine's daughter, who was a victim of her husband's sexual abuse of minor children.

Figure 1. This clip when viewed without audio would convey a person trying to kill another person and the police stopping it. But the description helps us to reason out why the woman is trying to kill the person.

In this work, we explore the problem of video highlight detection. Highlights in a video can fall into the following two categories: 1. Visually important: Shots of the video that are visually informative, and 2. Semantically important: Shots of the video that contribute to a higher level of understanding, which may or may not be visually good. A direct application of finding the highlights of the video is video summarization. Majority of the work done till date explore the concept of visual importance. The benchmark datasets [4], [5], specifically mute the video to obtain the visual importance only. The existing approaches for video summarization fall short considerably if only the visual importance is taken into account. For a better understanding of videos, there have been efforts to explore the combined relation between the video and text. But the task of capturing the semantics between the two is not trivial. Figure 1 is an example of a semantically important clip from the TV series *Numb3rs*. This can be considered the key clip of the episode, where the motive of the culprit is revealed in the dialogue/description. While the shot may be perceived as visually informative if muted, one will not be able to comprehend the true meaning as described in the text description. In this paper we propose a method to account for the semantically important shots in the domain of TV series where text/audio plays an key role.

The contributions of the paper are :

1) A method that utilizes video shot descriptions to identify highlights in the video. We use widely available extractive text summarization algorithms like TextRank [6], and LexRank [7] as ranking schemes

[1]http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html

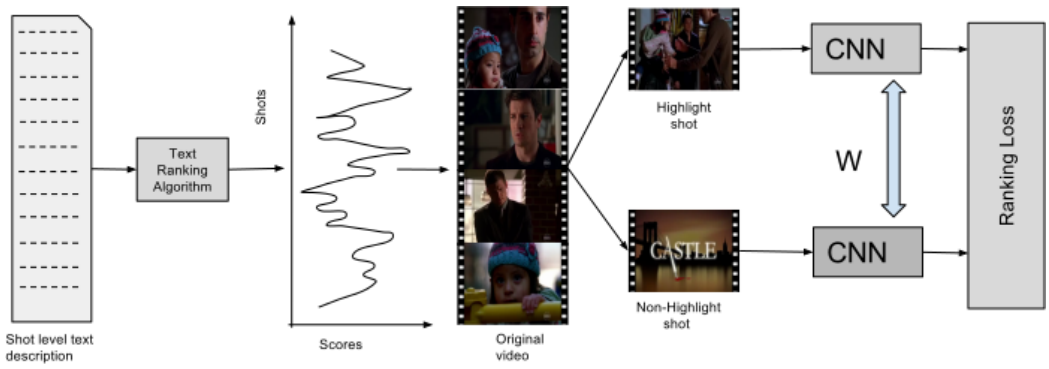

Figure 2. **Overview of the system.** We use an extractive text summarization algorithm to select descriptions and corresponding highlight shots of the video. A visual pairwise ranking model is trained using the highlights and the rest of the shots from the video to learn a score generating function.

for the importance of shots. Without loss of generality any text summarization algorithm can be used for this purpose.

2) A visual model trained on spatial and temporal information to obtain semantically important shots of a video. Note that the model does not need audio/text information from test videos.

We first look into the existing works of video highlight detection in Section II, before describing the proposed method in detail in Section III. Experimental results on different datasets and their analysis are presented in Section IV.

II. Related Work

A. Visual Content Summarization

The topic of finding interesting shots of a video was explored as event detection in the area of sports videos. [8], [9], [10].

Gygli [11] introduce video summarization problem as a subset selection problem and take into account video interestingness, representativeness, and uniformity of the frame/shot. A linear combination of the scores of these criteria is formed as a submodular function, which is then optimized. The results are presented as skims or frames that represent the video as a whole. The results are demonstrated on the egocentric dataset [12] and SumMe [4] dataset.

Sun [13] presented an idea where they harvest video highlights by analyzing raw and edited videos. They propose that shots appearing in the edited video retain highlights from the raw version. They formulate the problem using pairwise ranking constraints to learn a highlight detector. They evaluate on the YouTube[14] dataset which has around 100 videos of skating, surfing, skiing, gymnastics.

Yang [15] uses an unsupervised technique by training an auto encoder using domain specific web crawled short videos that act as highlights. They propose that an autoencoder that is trained on only highlights videos would be able to reconstruct highlights in the test scenario with higher accuracy, compared to a non-highlight shot. To achieve temporal dependence on preceding and subsequent frames they use an LSTM based recurrent autoencoder. Results are reported on the YouTube [14] dataset.

Lin [16] aims at summarizing egocentric videos that can be unstructured and varied in length. They propose to summarize during the recording to save memory and discard irrelevant parts. Their method involves an offline phase in which they learn a discriminative model for highlight detection and context prediction. During the online phase the video is uniformly partitioned into segments and processed to decide whether to retain or discard each segment. They model highlight detection and context prediction using a structured SVM in two formulations namely sequential and joint. They report the performance on the YouTube [14] dataset.

Yao [17] present a method for extracting video highlights in egocentric videos. They model the problem as a pairwise ranking model that generates a score for a video shot. The summary is constructed using a video timelapse (plays highlight shots at a slower speed and the non highlight shots at a faster frame rate) and a video skimming (top k scoring shots) techniques. A key contribution of the paper is that the summaries preserve continuity in a video sequence when compared to keyframe generating methods. [11],[18].

B. Semantic summarization methods

Recent works in the field that utilize both vision and language together are for the tasks of image captioning and query based video retrieval. For video summarization or highlight detection, combining visual criteria with semantic information extracted using pre-trained image-caption models have been explored.

Otani [18] used the video description dataset to create an embedding where semantics are preserved. The approach creates triplets using a positive pair (video and corresponding description) with the video acting as the anchor. The final summary is formulated as a k-mediod problem and the representative frames are chosen of the test video. The evaluation is done on the SumMe [4] dataset.

Sah [19] propose boundary detection of shots or superframes. They are then scored based on attention, colorfulness, contrast, face detection, etc. The keyframe detected

is then fed to a recurrent network to generate appropriate captions. One shortcoming of the method is that it fails to capture the names of the characters, in the caption generating model, hence does not perform well on the TV series dataset due to the lack of data (4 TV series episodes).

Note that all of these approaches require explicit generation of captions from the video or key frames to solve the problem of highlight detection. The proposed work in contrast does not depend on an automatic description generator as they do not work very well in domains such as TV series, where the audio information is complementary to the video and one cannot be inferred from the other. Hence we propose to use the actual audio content to determine semantic highlights and train a network to predict such semantically important shots using only the video content.

III. Method

We now describe our method below in the following stages. III-A Text ranking algorithms for score generation and III-B Pairwise shot ranking model.

Given an input video divided into N shots and its corresponding descriptions, our aim is to obtain the most relevant shots. Since multiple shots can be part of a scene, they can be assigned to the same description. In the rest of the paper text ranking is synonymous to test summarization.

A. Text Ranking

In order to mine the most important descriptions, we use a publicly available framework [2] for text summarization. For each of the input video for the TV series dataset, a 500 word summary is generated. A relevance score for each shot-description is computed according to the respective algorithm and the final summary is created using the Text MMR(Maximal Marginal Relevance) [20]. MMR chooses the representative descriptions and discards those with similarity greater than a specified threshold.

$$MMR \stackrel{\text{def}}{=} \arg\max_{s_i \in R \setminus S} [\lambda s_i - (1-\lambda) \max_{s_j \in S} Sim(s_i, s_j)] \quad (1)$$

When $\lambda = 1$ is the standard relevance list and when $\lambda = 0$ returns the most diverse sentences. R is the set of rank scores for each description of a video and S is the subset of R that contains already selected sentences for the summary.

ROUGE [3] is a widely used metric for the evaluation of text summaries. Rouge measures recall i.e. the overlap in the human reference summaries and in the machine generated summary. ROUGE SU reports the skip gram based F-measure and Recall of the machine generated and the reference summary. Based on the result shown in table I we use TextRank[6] algorithm in ranking descriptions. TextRank algorithm is a graph based ranking model for graphs made from natural language text. It assigns a score to each vertex of the graph with a relevance score. The edges of the graph denote the dependency of other sentences. Higher degree of a vertex represents high relevance. The edge weights of the graph is determined by the sentence similarity. The similarity function $Sim(s_i, s_j)$ between two sentences is computed using overlap of tokens. We normalize using the sentence length to avoid favouring selection of longer sentences. The text summary thus obtained is a subset of the original input. We assume that the sentences that are part of the summary refer to the highlights of the video. The rest of the descriptions form the non-highlights.

Table I
ROUGE-SU F-measure and Recall scores of GT vs Automatic summarization algorithms for TV04 of TV series VideoSet dataset

GT user	Centroid	ILP	LexRank	TextRank
GT1	R: 8.24	R: 8.2	R: 9.2	**R: 50.5**
	F: 8.9	F: 8.5	F: 9.8	**F: 52.9**
GT2	R: 32.9	R: 26.7	R: 7.3	**R: 24.4**
	F: 34.1	F: 26.2	F: 7.5	**F: 24.4**
GT3	R: 17.6	R: 7.6	R: 6.4	**R: 52.9**
	F: 17.9	F: 7.3	F: 6.4	**F: 52.0**

For the Buffy the Vampire Slayer TV series, the shot and text matching was done by Tapaswi [21] where the publicly available plot synopsis is aligned with best matching shots. These sentences are longer in length as compared to the Videoset dataset. The number of sentences per episode varies from 22 - 54 for a episode length of 45 min. Since these sentences are human-written and are well curated when compared to the VideoSet TV series data [12], TextRank generally chooses the first N sentences. In order to avoid this, we use a publicly available tool [4] that produces a importance heatmap of the given sentences. It generates more reliable summaries for this dataset.

B. Model Architecture

Yao [17] introduce a two stream architecture for highlight detection in a video. The two stream architecture consists of a spatial stream capturing important objects in a frame and the temporal stream that captures the temporal dynamics.

The input to the model are shots of the video, that can be obtained from a shot boundary detector, or by uniformly dividing the video in time. For pre-processing, we choose 5 non-overlapping clips of 2 secs at random intervals as per [22], to form the representation of the shot. From each clip, for the spatial information we extract 3 frames at uniform intervals. The spatial network thus has an input of 15 frames per shot. We extract features from pre-trained AlexNet [23] network which is trained on 1 million images

[2]https://github.com/PKULCWM/PKUSUMSUM
[3]https://pypi.python.org/pypi/pyrouge/0.1.3
[4]http://smmry.com/

and gives robust features. The features of the size 15×1000 are then passed through a average pooling layer to form a single representation of 1000 dimension. This is the input to fully connected layers with non-linearity, FC1000-FC512-FC256-FC128-FC64-FC1. We aim to learn a shot scoring function that assigns a higher score to a highlight shot compared to a non-highlight shot.

The temporal part of the model follows a similar architecture to the spatial, and we capture temporal information using 3D convolutions and 3D pooling from a sequence of frames. The features for each clip are extracted from pre-trained C3D [22] fc6 layer which has been trained on Sports 1M dataset [24]. Similar to the spatial network, average pooling is performed on the 5×4096 dimensional vectors per shot. Following which, we learn the score generating function with the fully connected layers FC4096-FC1024-FC512-FC256-FC128-FC1. In each of the networks, we set a dropout of 0.5. The non linear activation function after every learnable layer is Rectified Linear Unit (ReLU). We use late fusing of the scores to obtain the final scores. We train the spatial network and the temporal network keeping the same Train-Test split and fuse the scores to obtain the final score per shot. The training criterion is define, the highlight and the non-highlight scores output from the network should differ by a minimum margin of m. We use a margin of 1 for our experiments. Our objective is to penalize the violation of this criterion and minimize the loss over the training sample pairs.

$$s(h_i) > s(n_j) \quad \forall i \in H, \forall j \in N \tag{2}$$

$$min \quad L = \sum_{i \in H, j \in N} max(0, m - s(h_i) + s(n_j)) \tag{3}$$

IV. Experiments and Analysis

A. Dataset

Yeung [12] provide shots with dense text annotations of videos. We demonstrate out results on the TV Episodes of the datatset. There are 4 episodes of 45 minutes for each video. These videos are well edited and one advantage is that shots with blurring is not present. The videos are split uniformly into shots of 10 seconds each. The annotations for each shot is in third person. The tv01 episode is of *Castle* series, tv02 is an episode from *"The Mentalist"*, tv03 and tv04 belong to the *"Numb3rs"* series. This is the only video summarization dataset which provides text annotations. A total of 1036 shots are obtained and 263 (tv04) forms the test set.

We also evaluate our method on Buffy the Vampire slayer series which [21] which is a 22 episode series and is divided into shots using the a shot boundary detector. A total of 996 shots are obtained for all the episodes. Hence, the length of the shots will differ. We split the data randomly on episode basis with a split of 16-2-4 of Train-Validation-Test respectively.

Table II
Mean Average Precision values reported for the test data, tv04 of VideoSet and ep22 of Buffy dataset respectively. The empty fields indicate data not available for the dataset.

Method	mAP VideoSet	mAP Buffy
Uniform	0.1354	0.1958
Random	0.1418	0.2224
Spatial [17]	0.1941	-
Temporal [17]	0.2012	-
Spatial + Temporal [17]	0.1838	-
Spatial (Ours)	**0.2115**	**0.816**
Temporal (Ours)	**0.1976**	**0.8357**
Spatial + Temporal (Ours)	**0.2166**	**0.855**

B. Training

We used Torch 7 library for implementation and experimentation. The batch size was 16 and learning rate initialized to 1e-5 for the spatial part. For the temporal model the batch size is set to 8 and the learning rate is initialized to 1e-3. The optimizer used in the learning is Adam. A learning rate decay is set to 1e-7. A momentum parameter of 0.9 is set for both the models. The settings remain the same for both the datasets.

C. Evaluation

We compare our method with the baselines of sampling, namely uniform and random sampling of highlights in the test video. For comparison with [17], we collect visual annotations with the 3 human annotators. The annotations obtained for each clip is scored as one of the following (1) Boring - 1 (2) Normal - 2 (3) Highlight - 3. The shots which combined score of the annotators, greater than score value 8 is considered as highlights. We train the model on the obtained scores and compare them against the ground truth. We use Mean Average Precision (mAP) as a metric for evaluation. mAP is a metric commonly used in the retrieval domain which compares how close the ranking of the system are to the human assigned ranking. The ground truth is available for the the VideoSet [12] dataset both in text format and shot number format. We assign a value of 1 for the shots that are present in the summary and the rest are assigned a value of 0. For the Buffy dataset [21], since there is no groundtruth summary available we conducted an evaluation using 5 human evaluators to judge the quality of the summary.

Table II reports the mAP for both the datasets. The performance of the VideoSet dataset suffers due to the lack of data and also because the data comes from different TV series and the number of episodes is less in number too, hence it is difficult to capture the semantics of the data. We show the results on the TV 04 episode. We compare results of the spatial and the temporal models and the results are improves on use of a combined score of the spatial and temporal models. This conforms with our idea that both the spatial information and temporal dynamics contribute

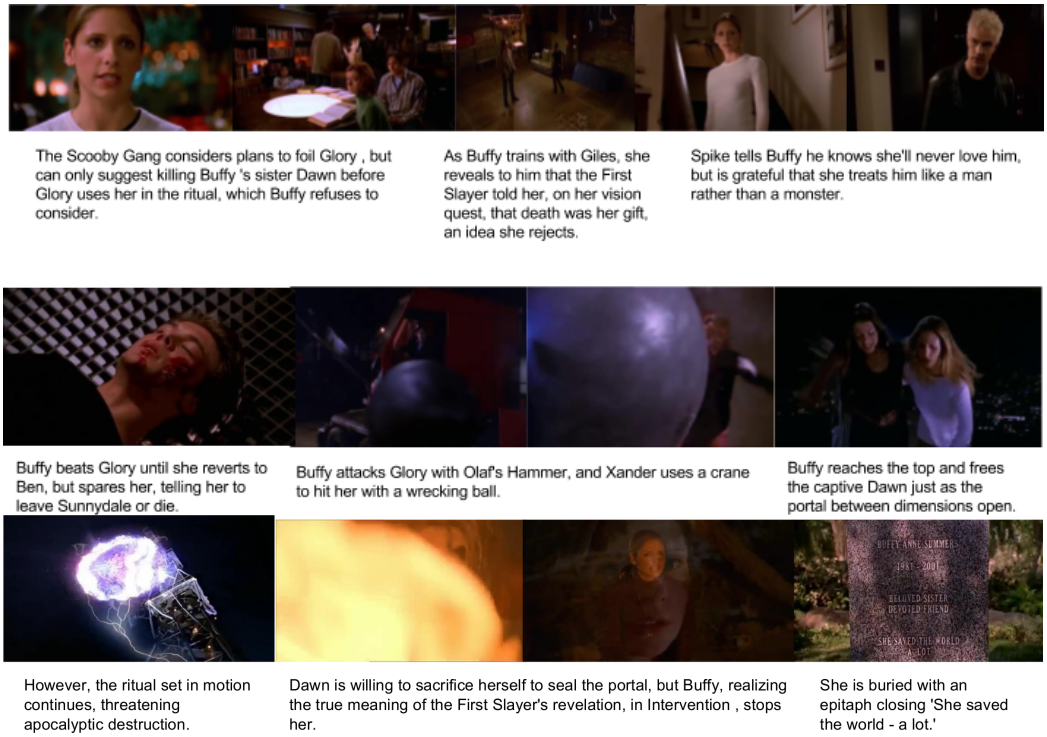

Figure 3. Example summary top scoring shots of Buffy episode 22 with the corresponding shot descriptions

to highlights. We combine the results of the spatial and the temporal model using a weighted sum denoted by

$$Score = \omega \times Spatial + (1 - \omega) \times Temporal \quad (4)$$

We compare the results with model trained on visual annotations obtained as per [17]. The results obtained with our method fall short by a margin of The results reported on Buffy dataset perform better than the VideoSet data, in which the characters, places remains constant and the model is able to learn on this dataset. We see a significant improvement in the results for this dataset.

We set the value of $\omega = 0.3$ for all the experiments as per [17].

For human evaluation, for the Buffy datatset, we create a summary using the top k scoring shots. Since the shots vary in length, we increase the frame rate to create summary of 1 minute in length. To convey the context we display the shot descriptions along with the shots. The criterion of the human evaluation was presentation and coverage of the original video as per [17]. The users were asked to rate the summary on the scale of 1 (Poor) - 5(Excellent) for the Buffy dataset. The 5 evaluators were asked to evaluate on the basis of viewing the episode and then view summary video. They reported a consistent rating of 3.5 for both the criterion. Some of the shortcomings reported was that there was no audio to summary generated, the longer shots dominate the summary and the smaller shots cannot convey the context when displayed at a higher frame rate. In 3 we demonstrate the qualitative results for the episode 22 from the dataset.

V. Future work and Conclusion

In this paper we explored a method that utilizes text descriptions of shots to find the highlights in a video. The proposed method reduces the effort of visual annotations. The results can be improved upon by using better text ranking algorithms and sufficient amount of video data. The learned score generating function can be used for future architectures as a video highlight detector. We require text rich annotated shots for our purpose, which requires aligning the text and shots. A future direction of work is using more commonly available subtitles for highlight detection. Also, choosing the best representative sub shots rather than the long shot can be explored while maintaining continuity, just like normalizing longer sentences in text ranking.

References

[1] C. Xu, Y.-F. Zhang, G. Zhu, Y. Rui, H. Lu, and Q. Huang, "Using webcast text for semantic event detection in broadcast sports video," *IEEE Transactions on Multimedia*, vol. 10, no. 7, pp. 1342–1355, 2008. 1

[2] L. Wang, Y. Li, and S. Lazebnik, "Learning deep structure-preserving image-text embeddings," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5005–5013. 1

[3] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, "Sequence to sequence – video to text," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015. 1

[4] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *ECCV*, 2014. 1, 2

[5] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "Tvsum: Summarizing web videos using titles," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5179–5187. 1

[6] R. Mihalcea and P. Tarau, "TextRank: Bringing order into texts," in *Proceedings of EMNLP-04and the 2004 Conference on Empirical Methods in Natural Language Processing*, July 2004. 1, 3

[7] G. Erkan and D. R. Radev, "LexRank: Graph-based lexical centrality as salience in text summarization," *Journal of Artificial Intelligence Research*, vol. 22, pp. 457–479, 2004. 1

[8] D. Yow, B.-L. Yeo, M. Yeung, and B. Liu, "Analysis and presentation of soccer highlights from digital video," in *proc. ACCV*, vol. 95, 1995, pp. 499–503. 2

[9] L. Ballan, M. Bertini, A. Del Bimbo, L. Seidenari, and G. Serra, "Event detection and recognition for semantic annotation of video," *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 279–302, 2011. 2

[10] B. Li and M. I. Sezan, "Event detection and summarization in sports video," in *Proceedings IEEE Workshop on Content-Based Access of Image and Video Libraries (CBAIVL 2001)*, 2001, pp. 132–138. 2

[11] M. Gygli, H. Grabner, and L. Van Gool, "Video summarization by learning submodular mixtures of objectives," in *CVPR*, 2015. 2

[12] S. Yeung, A. Fathi, and L. Fei-Fei, "Videoset: Video summary evaluation through text," *CoRR*, vol. abs/1406.5824, 2014. [Online]. Available: http://arxiv.org/abs/1406.5824 2, 3, 4

[13] M. Sun, A. Farhadi, T. H. Chen, and S. Seitz, "Ranking highlights in personal videos by analyzing edited videos," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5145–5157, Nov 2016. 2

[14] M. Sun, A. Farhadi, and S. Seitz, "Ranking domain-specific highlights by analyzing edited videos," in *ECCV*, 2014. 2

[15] H. Yang, B. Wang, S. Lin, D. Wipf, M. Guo, and B. Guo, "Unsupervised extraction of video highlights via robust recurrent auto-encoders," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 4633–4641. 2

[16] Y.-L. Lin, V. I. Morariu, and W. Hsu, "Summarizing while recording: Context-based highlight detection for egocentric videos," *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, vol. 00, pp. 443–451, 2015. 2

[17] T. Yao, T. Mei, and Y. Rui, "Highlight detection with pairwise deep ranking for first-person video summarization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 982–990. 2, 3, 4, 5

[18] M. Otani, Y. Nakashima, E. Rahtu, J. Heikkilä, and N. Yokoya, *Video Summarization Using Deep Semantic Features*. Cham: Springer International Publishing, 2017, pp. 361–377. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-54193-8_23 2

[19] S. Sah, S. Kulhare, A. Gray, S. Venugopalan, E. Prud'Hommeaux, and R. Ptucha, "Semantic text summarization of long videos," in *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*. IEEE, 2017, pp. 989–997. 2

[20] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1998, pp. 335–336. 3

[21] M. Tapaswi, M. Bäuml, and R. Stiefelhagen, "Story-based video retrieval in tv series using plot synopses," in *Proceedings of International Conference on Multimedia Retrieval*, ser. ICMR '14. New York, NY, USA: ACM, 2014, pp. 137:137–137:144. [Online]. Available: http://doi.acm.org/10.1145/2578726.2578727 3, 4

[22] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 3, 4

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf 3

[24] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '14. Washington, DC, USA: IEEE Computer Society, 2014, pp. 1725–1732. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2014.223 4