Leveraging Audio and Visual Recurrence for Unsupervised Video Highlight Detection

Zahidul Islam[1] **Sujoy Paul**[2] **Mrigank Rochan**[1]
[1]University of Saskatchewan, Canada [2]Google DeepMind

Abstract

With the exponential growth of video content, the need for automated methods to extract key moments or highlights from lengthy videos has become increasingly pressing. Existing methods typically require expensive manually labeled annotations or a large external dataset for weak supervision. Hence, we propose a novel unsupervised approach which capitalizes on the premise that significant moments tend to recur across multiple videos of the similar category in both audio and visual modalities. Surprisingly, audio remains under-explored, especially in unsupervised algorithms, despite its potential to detect key moments. Our approach first groups videos into pseudo-categories using a clustering technique. Then, by measuring clip-level feature similarities across all videos within each pseudo-category for both audio and visual modalities, we obtain audio and visual pseudo-highlight scores, respectively. We combine these scores to create audio-visual pseudo ground-truth highlights for each video, which we subsequently use to train an audio-visual highlight detection network. Extensive experiments and ablation studies on three benchmarks show the superior performance of our method compared to prior work.

1 Introduction

Video highlight detection, a critical task in the realm of video content analysis, aims to automatically identify the most important or engaging segments from lengthy video content [1, 2, 13, 44]. As the volume of video data on the internet continues to surge, there is a growing demand for efficient methods to navigate and consume such content with applications ranging from sports broadcasting to content creation, education, marketing, surveillance, entertainment, and beyond. Supervised approaches for highlight detection are popular but require expensive manually annotated frame-level supervision [1, 15, 43, 47, 12, 15, 34, 23]. To address this, there is a stream of research on weakly supervised learning [44, 46, 3, 13, 29], which utilizes video-level labels such as video category as a weak supervision signal. However, they typically require a large external dataset, such as web-crawled videos, for model training. In light of this, the largely unexplored development of unsupervised algorithms are promising for automated highlight detection without any labels. Hence, we introduce an innovative unsupervised method which leverages cues from both audio and visual components of the video to improve video highlight detection. Interestingly, audio cues are often overlooked, but they can be highly informative for highlight detection. To the best of our knowledge, our work represents the first attempt to exploit both audio and visual components in unsupervised learning for video highlight detection without requiring any large external dataset.

Videos with similar content or actions, tend to exhibit recurrence of key moments in both the audio and visual modalities. By *recurrence*, we mean the repetition of specific patterns or features in multiple videos of similar categories. These may manifest as audio cues, like the repeated occurrence of specific sounds or phrases, or as visual cues, such as the reappearance of particular objects or scenes. For example, in cooking videos, close-up shots of food depicting certain actions appear frequently, such as chopping vegetables, stirring ingredients, or the sizzling sound of food being cooked.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

Similarly, in sports videos, cheering of the crowd or the excitement in the commentator's voice may recur as audio cues, while the slow-motion replay of a goal or a player's celebration may recur as visual cues, both signaling key moments for highlight. We also visualize these observations in Figure 1. In this work, we combine the strengths of both auditory and visual cues to detect highlights through their inherent recurrence.

We propose an unsupervised algorithm to identify and leverage these recurrences and generate a supervisory signal in the form of audio-visual pseudo-highlights for training a highlight detection network end-to-end. During testing, we concatenate the top-scoring clips of an input video to generate its highlight. The proposed unsupervised method not only outperforms state-of-the-art unsupervised methods on benchmark highlight detection datasets but also demonstrates comparable or superior performance to state-of-the-art weakly supervised methods.

Figure 1: In *cooking* videos, close-up shots of food depicting various actions (chopping, pan-frying) are recurring highlight moments (1st and 2nd rows). In *gymnastics* videos, recurring audio cues, such as cheers and claps, occur when spectators react to interesting acrobatic moves (3rd and 4th rows).

2 Our Approach

Let's say we have a set of M unlabeled videos $\{V_j\}_{j=1}^{M}$. We split each video V_j into clips with an equal number of frames, resulting in n clips. From each clip c_i (where $i = 1, 2, ...n$), we extract its corresponding visual features $v_i \in \mathbb{R}^{d_v}$ using a pre-trained visual feature extractor and audio features $a_i \in \mathbb{R}^{d_a}$ using a pre-trained audio feature extractor. Each video V_j can then be represented using its set of visual features, $\{v_i\}_{i=1}^{n}$, and audio features, $\{a_i\}_{i=1}^{n}$. Our highlight detection method predicts a set of highlight scores $\{h_i\}_{i=1}^{n}$, where h_i indicates the highlight score of each clip c_i in video V_j. Next, we discuss the three broad steps in our approach as illustrated in Figure 2.

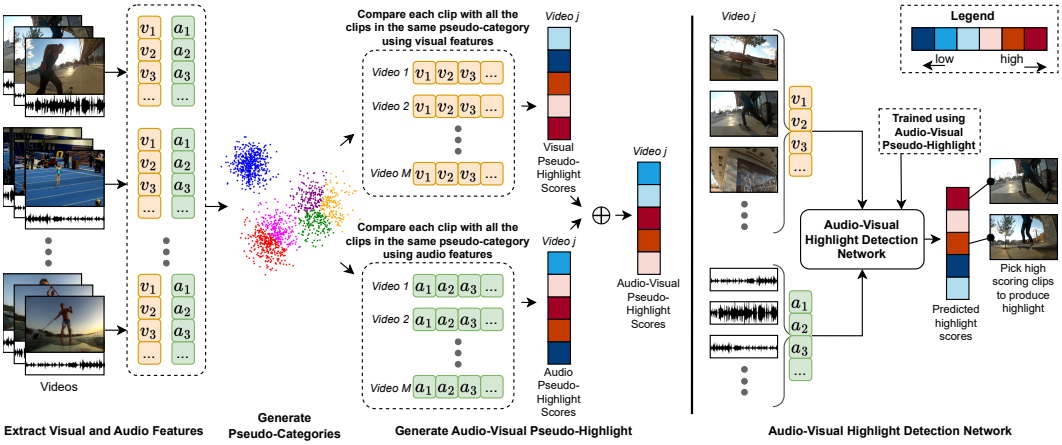

Figure 2: An overview of our unsupervised video highlight detection framework.

Generating Pseudo-Categories: In this step, we assign the unlabeled videos to multiple groups using a clustering-based approach. We refer to these groups as *pseudo-categories*. From each video V_j in the training set of a dataset, we calculate the mean of its clip-level visual features $\{v_i\}_{i=1}^{n}$ as $\bar{v} = \frac{1}{n}\sum_{i=1}^{n} v_i$. Similarly, we calculate the mean of its clip-level audio features $\{a_i\}_{i=1}^{n}$ as $\bar{a}$. Subsequently, we concatenate $\bar{v}$ and $\bar{a}$ to obtain a video-level audio-visual feature representation $\bar{f} = [\bar{v}; \bar{a}]$ of the video, which we utilize for clustering. We reduce the dimensionality of $\bar{f}$ using UMAP [26], resulting in transformed $\bar{f} \in \mathbb{R}^{10}$, which is a common pre-processing step in clustering

literature [37, 25]. Following standard practice for finding an optimal number of clusters, we iteratively apply the K-means clustering algorithm using the transformed features $\bar{f}$ of the videos for a range of values for K. For each K, we calculate a standard clustering fitness metric, the Silhouette Coefficient (SC) [38], and choose the K with the highest SC as the optimal number of clusters. We use the cluster labels assigned by clustering with the optimal K as our pseudo-categories for the videos. These pseudo-categories are generated by clustering video-level audio-visual features extracted from pre-trained classification models, potentially grouping together videos with similar semantics. Next, we utilize these pseudo-categories when generating the audio-visual pseudo-highlight for each video.

Generating Audio-Visual Pseudo-Highlight: For each video, we first compare each of its clips with all the clips across videos of the same pseudo-category using their audio and visual features to obtain audio pseudo-highlight and visual pseudo-highlight scores, respectively. We then aggregate these scores to generate audio-visual pseudo-highlight of the video. Let, there are K videos assigned to a particular pseudo-category, and $\{a_k\}_{k=1}^{S}$ represents a set of audio features corresponding to all the S clips across all K videos of that pseudo-category. We assign an audio pseudo-highlight score aph_i to the i-th clip of a video based on how repetitive the corresponding audio features a_i are in the videos of that pseudo-category (referred to as audio recurrence). We use cosine similarity to compare the audio features of a clip a_i in a video with all the clips in the pseudo-category. We can write aph_i computation as, $aph_i = \frac{1}{S}\sum_{k=1}^{S} a_i \cdot a_k / \|a_i\| \|a_k\|$. Similarly, we also compute a visual pseudo-highlight score vph_i for the i^{th} clip of a video based on the similarity of the corresponding visual features v_i in that pseudo-category (referred to as visual recurrence): $vph_i = \frac{1}{S}\sum_{k=1}^{S} v_i \cdot v_k / \|v_i\| \|v_k\|$. Finally, for each clip in a video, we compute the average of audio and visual pseudo-highlight scores and select the top $t\%$ clips based on the average scores to obtain the audio-visual pseudo-highlight (AV-PH) of the video, which we use as the supervision to train our highlight detection network.

Audio-Visual Highlight Detection Network: We adopt the audio-visual highlight detection network from [1], which has been shown to be highly effective for supervised highlight detection. The network is optimized by minimizing the binary cross-entropy loss between the predicted highlight scores and the generated audio-visual pseudo-highlights (AV-PH). Further details about the network architecture and its training can be found in Appendix A.2.

3 Experiments

Datasets and Comparison Methods: We evaluate our method using three benchmark video highlight detection datasets: YouTube Highlights [41], TVSum [40], and QVHighlights [21]. On YouTube and TVSum, we compare our approach with state-of-the-art weakly supervised methods, including TC [46], MN [13], LM[44], RRAE [45], MBF [4], CVS [30], DSN [29], and VESD [3], as well as unsupervised methods, including SG [24], BT [39], CHD [2], and MT [22]. Among the compared methods, MN, TC, and MT utilize both visual and audio modalities. On QVHighlights, we compare with unsupervised approaches that do not utilize textual queries: BT [39] and CHD [2].

Main Results: In Table 1, we compare the performance of our approach on YouTube. Without any weak supervision or external dataset for training, our unsupervised approach not only outperforms the prior methods but also improves the state-of-the-art by almost 3%. On QVHighlights, our approach

Method	RRAE	LM	MN	TC	MT	CHD	Ours
Ext. data	✓	✓	✓	✓	✓	✗	✗
mAP	38.30	56.40	61.38	62.97	65.10	65.39	**68.30**

Table 1: Highlight detection results on YouTube. Our approach outperforms all prior unsupervised and weakly supervised methods.

Method	BT	CHD	Ours
mAP	14.36	15.82	**18.38**
HIT@1	20.88	17.10	**24.71**

Table 2: Highlight detection results on the QVHighlights *test* split from the QVHighlights evaluation server.

significantly outperforms (see Table 2) both of the prior unsupervised methods BT and CHD which, similar to our work, do not utilize query information. CHD does not evaluate on QVHighlights, so we implement and evaluate their method on this dataset to compare with our approach. In Table 3, we compare our approach on TVSum with unsupervised and weakly supervised methods that do not rely on an external dataset for training. Again, we achieve state-of-the-art performance outperforming the best prior method CHD by about 8%. Moreover, in Table 4, we compare with weakly supervised

Method	MBF	CVS	SG	DSN	CHD	Ours
Ext. data	✗	✗	✗	✗	✗	✗
t-5 mAP	34.50	37.20	46.20	42.40	52.76	**60.34**

Table 3: Results results on TVSum. We compare with existing unsupervised and weakly supervised methods that do not rely on external data.

Method	VESD	LM	MN	TC	MT	Ours
Ext. data	✓	✓	✓	✓	✓	✗
t-5 mAP	42.30	56.30	73.24	76.82	78.30	60.34

Table 4: Comparison with prior works that utilize external data on TVSum. Even without external data, our method exceeds three of these methods.

Method	TVSum	YouTube	QVHighlights *val*	
	t-5 mAP	mAP	mAP	HIT@1
AV(V-PH)	53.35	62.76	17.58	20.19
AV(A-PH)	58.34	66.36	17.81	23.10
Ours	**60.34**	**68.30**	**18.41**	**26.26**

Table 5: Analyzing the impact of audio and visual modality on generating pseudo-highlights.

Method	TVSum	YouTube	QVHighlights *val*	
	t-5 mAP	mAP	mAP	HIT@1
Pseudo-highlight	52.59	58.71	17.60	23.61
Ours	**60.34**	**68.30**	**18.41**	**26.26**

Table 6: Evaluating quality of audio-visual pseudo-highlights as supervision for highlight detection.

methods on TVSum that require a large external dataset of web-crawled videos for training. Even without the advantage of external data, our method outperforms three of these methods.

Ablation Studies: We examine the impact of each modality and their combination for pseudo-highlight generation in Table 5. The audio-visual highlight detection model AV, trained solely with audio pseudo-highlights (A-PH), outperforms the version trained with visual pseudo-highlights (V-PH). This indicates that recurring moments in the audio can provide strong cues for detecting highlights, emphasizing the significance of audio modality. The results also show combined audio-visual pseudo-highlights (AV-PH) are more effective than using either modality alone, as these two modalities can contain complementary information about potential highlight moments. Additionally, to evaluate the quality of generated pseudo-highlights as supervision, we use the audio-visual pseudo-highlight scores of each video as predictions and compare them (see Table 6) with the predictions of our highlight detection model trained on audio-visual pseudo-highlights. Despite potential noise in the pseudo-highlight supervision, our model learns from it during training, demonstrating its value and achieving better performance than using the pseudo-highlights directly as predictions. We provide additional experiment details in Appendix A.3 and ablation studies in Appendix A.4.

Qualitative Results: We visualize the prediction of our method on an example test video from TVSum in Figure 3. The video shows the preparation of a sandwich, with mostly close-up shots of food indicated as highlights. Our model correctly identifies most of the highlight moments.

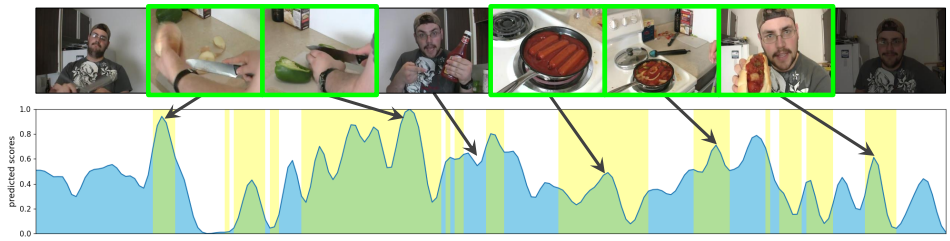

Figure 3: Qualitative results on a test video of *sandwich preparation* from TVSum. Highlight clips (in green) are shown with our predicted scores (in blue) and ground truth regions (in yellow).

4 Conclusion

We introduce a novel unsupervised audio-visual highlight detection framework. Our core idea is based on the premise that videos of similar categories tend to contain key moments that are repetitive in both audio and visual modalities across multiple videos. Leveraging this observation, we construct audio-visual pseudo-highlights to train our model. Extensive experiments showcase the effectiveness of our framework and reveal that cues from the often overlooked yet informative audio modality, when coupled with the visual modality, lead to a significant improvement in unsupervised learning of video highlight detection.

References

[1] Taivanbat Badamdorj, Mrigank Rochan, Yang Wang, and Li Cheng. Joint visual and audio learning for video highlight detection. In *IEEE/CVF International Conference on Computer Vision*, 2021.

[2] Taivanbat Badamdorj, Mrigank Rochan, Yang Wang, and Li Cheng. Contrastive learning for unsupervised video highlight detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[3] Sijia Cai, Wangmeng Zuo, Larry S Davis, and Lei Zhang. Weakly-supervised video summarization using variational encoder-decoder and web prior. In *European Conference on Computer Vision*, 2018.

[4] Wen-Sheng Chu, Yale Song, and Alejandro Jaimes. Video co-summarization: Video summarization by visual co-occurrence. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[5] Israel Cohen, Yiteng Huang, Jingdong Chen, Jacob Benesty, Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson correlation coefficient. *Noise Reduction in Speech Processing*, pages 1–4, 2009.

[6] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. Summarizing videos with attention. In *Computer Vision–ACCV 2018 Workshops: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers 14*, pages 39–54, 2019.

[7] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *IEEE/CVF International Conference on Computer Vision*, 2019.

[8] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

[9] Boqing Gong, Wei-Lun Chao, Kristen Grauman, and Fei Sha. Diverse sequential subset selection for supervised video summarization. *Advances in Neural Information Processing Systems*, 2014.

[10] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos. In *European Conference on Computer Vision*, 2014.

[11] Michael Gygli, Helmut Grabner, and Luc Van Gool. Video summarization by learning submodular mixtures of objectives. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[12] Michael Gygli, Yale Song, and Liangliang Cao. Video2gif: Automatic generation of animated gifs from video. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[13] Fa-Ting Hong, Xuanteng Huang, Wei-Hong Li, and Wei-Shi Zheng. Mini-net: Multiple instance ranking network for video highlight detection. In *European Conference on Computer Vision*, 2020.

[14] Tzu-Chun Hsu, Yi-Sheng Liao, and Chun-Rong Huang. Video summarization with spatiotemporal vision transformer. *IEEE Transactions on Image Processing*, 32:3013–3026, 2023.

[15] Yifan Jiao, Xiaoshan Yang, Tianzhu Zhang, Shucheng Huang, and Changsheng Xu. Video highlight detection via deep ranking modeling. In *Image and Video Technology: 8th Pacific-Rim Symposium, PSIVT 2017, Wuhan, China, November 20-24, 2017, Revised Selected Papers 8*, pages 28–39, 2018.

[16] Aditya Khosla, Raffay Hamid, Chih-Jen Lin, and Neel Sundaresan. Large-scale video summarization using web-image priors. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[17] Gunhee Kim and Eric P Xing. Reconstructing storyline graphs for image recommendation from web community photos. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[19] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894, 2020.

[20] Yong Jae Lee, Joydeep Ghosh, and Kristen Grauman. Discovering important people and objects for egocentric video summarization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[21] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. In *Advances in Neural Information Processing Systems*, 2021.

[22] Tingtian Li, Zixun Sun, and Xinyu Xiao. Unsupervised modality-transferable video highlight detection with representation activation sequence learning. *IEEE Transactions on Image Processing*, 2024.

[23] Ye Liu, Siyuan Li, Yang Wu, Chang-Wen Chen, Ying Shan, and Xiaohu Qie. Umt: Unified multi-modal transformers for joint video moment retrieval and highlight detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[24] Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. Unsupervised video summarization with adversarial lstm networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[25] Ryan McConville, Raul Santos-Rodriguez, Robert J Piechocki, and Ian Craddock. N2d:(not too) deep clustering via clustering the local manifold of an autoencoded embedding. In *International Conference on Pattern Recognition*, 2021.

[26] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[27] WonJun Moon, Sangeek Hyun, SangUk Park, Dongchan Park, and Jae-Pil Heo. Query-dependent video representation for moment retrieval and highlight detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[28] Medhini Narasimhan, Arsha Nagrani, Chen Sun, Michael Rubinstein, Trevor Darrell, Anna Rohrbach, and Cordelia Schmid. Tl; dw? summarizing instructional videos with task relevance and cross-modal saliency. In *European Conference on Computer Vision*, 2022.

[29] Rameswar Panda, Abir Das, Ziyan Wu, Jan Ernst, and Amit K Roy-Chowdhury. Weakly supervised summarization of web videos. In *IEEE International Conference on Computer Vision*, 2017.

[30] Rameswar Panda and Amit K Roy-Chowdhury. Collaborative summarization of topic-related videos. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[31] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, 2019.

[32] Danila Potapov, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. Category-specific video summarization. In *European Conference on Computer Vision*, 2014.

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

[34] Mrigank Rochan, Mahesh Kumar Krishna Reddy, Linwei Ye, and Yang Wang. Adaptive video highlight detection by learning from user history. In *European Conference on Computer Vision*, 2020.

[35] Mrigank Rochan and Yang Wang. Video summarization by learning from unpaired data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[36] Mrigank Rochan, Linwei Ye, and Yang Wang. Video summarization using fully convolutional sequence networks. In *European Conference on Computer Vision*, 2018.

[37] Meitar Ronen, Shahaf E Finder, and Oren Freifeld. Deepdpm: Deep clustering with an unknown number of clusters. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[38] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[39] Yale Song, Miriam Redi, Jordi Vallmitjana, and Alejandro Jaimes. To click or not to click: Automatic selection of beautiful thumbnails from videos. In *ACM International on Conference on Information and Knowledge Management*, 2016.

[40] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[41] Min Sun, Ali Farhadi, and Steve Seitz. Ranking domain-specific highlights by analyzing edited videos. In *European Conference on Computer Vision*, 2014.

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

[43] Lezi Wang, Dong Liu, Rohit Puri, and Dimitris N Metaxas. Learning trailer moments in full-length movies with co-contrastive attention. In *European Conference on Computer Vision*, 2020.

[44] Bo Xiong, Yannis Kalantidis, Deepti Ghadiyaram, and Kristen Grauman. Less is more: Learning highlight detection from video duration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[45] Huan Yang, Baoyuan Wang, Stephen Lin, David Wipf, Minyi Guo, and Baining Guo. Unsupervised extraction of video highlights via robust recurrent auto-encoders. In *IEEE International Conference on Computer Vision*, 2015.

[46] Qinghao Ye, Xiyue Shen, Yuan Gao, Zirui Wang, Qi Bi, Ping Li, and Guang Yang. Temporal cue guided video highlight detection with low-rank audio-visual fusion. In *IEEE/CVF International Conference on Computer Vision*, 2021.

[47] Youngjae Yu, Sangho Lee, Joonil Na, Jaeyun Kang, and Gunhee Kim. A deep ranking model for spatio-temporal highlight detection from a 360 video. In *AAAI Conference on Artificial Intelligence*, 2018.

[48] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Summary transfer: Exemplar-based subset selection for video summarization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[49] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In *European Conference on Computer Vision*, 2016.

[50] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. Hierarchical recurrent neural network for video summarization. In *ACM International Conference on Multimedia*, 2017.

[51] Kaiyang Zhou, Yu Qiao, and Tao Xiang. Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. In *AAAI Conference on Artificial Intelligence*, 2018.

A Appendix

A.1 Related Work

Video Highlight Detection: Most of the existing approaches for highlight detection rely on manually annotated frame-level supervision [12, 15, 34, 43, 47, 1, 23]. However, these annotations are laborious and expensive to obtain. To address this limitation, some works focus on using only video-level tags or category information as weak supervision [45, 3, 13, 29, 44, 46]. Several works adopt ranking frameworks. For example, LM [44] exploits video duration and ranks clips from shorter videos higher, whereas, Ye *et al.* [46] leverages temporal reasoning and encodes cross-modal relationships using an efficient audio-visual tensor fusion mechanism. However, these methods require training on large-scale external data. A recent unsupervised method based on contrastive learning [2] does not utilize any external data. However, they do not exploit audio for unsupervised learning. Some recent works explore supervised query-based highlight detection, where the goal is to extract highlight relevant to a given textual query [21, 23, 27]. In contrast, we focus on unsupervised audio-visual highlight detection which does not rely on any text input.

Video Summarization: When summarizing videos, the aim is to generate a coherent and concise synopsis of the video, whereas video highlight detection aims to extract significant moments. Earlier works on video summarization utilize unsupervised heuristics such as representativeness and diversity of the selected clips [16, 17, 39, 20, 24, 30, 40, 51]. Some methods use only weak video-level supervision [3, 29, 32, 35], while others employ supervised learning [9, 10, 11, 36, 48, 49, 50]. Recent works focus on capturing contextual dependencies using recurrent networks [49, 50] or attention layers [6, 14]. Our work is partly related to a recent study on instructional video summarization [28], which uses visual data and textual transcripts to construct pseudo-summaries comprised of the most salient steps to train their model. However, they do not exploit audio modality and require category information of videos for weak supervision.

A.2 Audio-Visual Highlight Detection Network

We adopt the network architecture from prior work [1] for our audio-visual highlight detection network (AV), a network design that has been proven to be highly effective in supervised highlight detection. At its core, our network initially employs unimodal self-attention layers [42] to capture clip-level temporal relationships within each modality using their features. Subsequently, these self-attended visual and audio features are fed into bimodal cross-attention layers to encode cross-modal dependencies and produce bimodal attended features. Finally, the self-attended features and bimodal attended features are combined and forwarded to fully-connected layers to predict the highlight score of each clip in the video.

More concretely, given a video with n clips, our audio-visual model processes the clip-level visual features $\{v_i\}_{i=1}^n$ using a self-attention layer $\text{Attn}_{v\to v}$ and the clip-level audio features $\{a_i\}_{i=1}^n$ using another self-attention layer $\text{Attn}_{a\to a}$. Then, two bimodal attention layers $\text{Attn}_{v\to a}$ and $\text{Attn}_{a\to v}$ process the self-attended visual features $\{v_i^v\}_{i=1}^n$ and self-attended audio features $\{a_i^a\}_{i=1}^n$ to produce bimodal attended features, $\{v_i^a\}_{i=1}^n$ and $\{a_i^v\}_{i=1}^n$, respectively. Finally, a score regressor module (SR) combines self-attended and bimodal attended features using learnable weights and passes them through two fully-connected layers to predict the highlight score h_i for each clip in the video. These operations can be expressed as follows:

$$\{v_i^v\}_{i=1}^n = \text{Attn}_{v\to v}(\{v_i\}_{i=1}^n) \tag{1}$$

$$\{a_i^a\}_{i=1}^n = \text{Attn}_{a\to a}(\{a_i\}_{i=1}^n) \tag{2}$$

$$\{a_i^v\}_{i=1}^n = \text{Attn}_{a\to v}(\{a_i^a\}_{i=1}^n, \{v_i^v\}_{i=1}^n) \tag{3}$$

$$\{v_i^a\}_{i=1}^n = \text{Attn}_{v\to a}(\{v_i^v\}_{i=1}^n, \{a_i^a\}_{i=1}^n) \tag{4}$$

$$\{h_i\}_{i=1}^n = \text{SR}(\{v_i^v\}_{i=1}^n, \{a_i^a\}_{i=1}^n, \{v_i^a\}_{i=1}^n, \{a_i^v\}_{i=1}^n) \tag{5}$$

The training procedure for our network follows the standard approach used in the supervised network [1]. However, instead of using ground-truth annotations, we use audio-visual pseudo-highlights as the supervisory signal to train our network. We optimize our network by minimizing the binary cross-entropy loss between the predicted highlight scores and the audio-visual pseudo-highlights (AV-PH).

A.3 Datasets and Settings

We evaluate our method using three benchmark highlight detection datasets: YouTube Highlights [41], TVSum [40], and QVHighlights [21]. YouTube Highlights is constructed by mining YouTube videos related to six specific categories such as parkour, gymnastics, skiing, and so on, with about 100 videos in each category. We utilize the train and test splits provided in this dataset. TVSum has 50 videos across 10 diverse categories. Following prior works [1, 2, 35], we randomly split this dataset with 80% of the videos for training and 20% for testing, and we run our experiments on this dataset five times and report the average performance. QVHighlights is larger with over 10,000 videos. It is primarily designed for query-focused video highlight detection and moment retrieval. Each video is associated with a textual query and corresponding saliency/highlight scores. The dataset comes with standard train, validation, and test splits with a ratio of 70:15:15. Since our method only requires videos, we ignore the user query annotations. For a fair comparison, on QVHighlights, we evaluate our method against prior non-query-based methods.

Features: For YouTube and TVSum, we follow prior work [2, 1] and use a 3D-CNN comprised of a ResNet-34 backbone to extract visual features from each clip. For QVHighlights dataset, following [21, 23], we extract visual features using SlowFast [7] and video encoder of CLIP (ViT-B/32) [33]. For all datasets, we employ PANN [19] audio network pre-trained on AudioSet [8] to extract audio features.

Evaluation Metrics: Following prior works on QVHighlights [23, 21], we report our performance using Mean Average Precision (mAP), which considers the highlight scores for all the clips, and HIT@1, which considers the hit ratio of the clip with the highest score for each video. We consider only the clips rated as *Very Good* by users for evaluation. On YouTube, we evaluate using mAP, and on TVSum, we report mAP on the top five predicted clips (top-5 mAP) as in prior works [2, 1]. All metrics are reported as percentages.

Implementation Details: We implement our models using PyTorch [31]. For YouTube and TVSum, we utilize one self-attention and one bi-modal attention module in our audio-visual highlight detector network, as in the previous method [1]. However, since QVHighlights is a much larger dataset, following previous studies [23], we introduce an additional self-attention module and fully connected layer in the score regressor to handle the increased complexity and scale of the data. We use Adam to optimize [18] our models. We train our models for 20 epochs with a learning rate of 2.5×10^{-3} on TVSum and for 100 epochs with a learning rate of 5×10^{-3} on YouTube. For QVHighlights, we train our models with a learning rate of 5×10^{-4} for 10 epochs. As mentioned in Sec. 2, we empirically select the number of clusters, K, by maximizing the Silhouette Coefficient. For all three datasets, we search in the range of 4 to 15 to find the value of K. For YouTube, TVSum, and QVHighlights, we find the optimal values of K to be 6, 8, and 7, respectively. Following prior works [2, 13], we select the top $t = 50\%$ clips based on the audio-visual pseudo-highlight scores (AV-PH) to create pseudo-highlights for training.

A.4 Additional Ablation Studies

We conduct extensive ablation studies to analyze the relative impact of each component of our approach. For QVHighlights, we follow prior work [21, 23] and utilize its *val* split due to limited number of submissions to the evaluation server. For TVSum and YouTube, we use the *test* split. We define the following baselines for our experiments.

- **A, V, and AV:** A and V denote unimodal models with a single self-attention layer that is trained on only audio and visual features, respectively. AV denotes the audio-visual highlight detection model in our proposed method.
- **A-PH, V-PH, and AV-PH:** A-PH refers to audio-visual highlight detection model that is trained using only audio pseudo-highlights. V-PH indicates the model trained using only visual pseudo-highlights. Finally, AV-PH, trained using audio-visual pseudo-highlights, is the proposed model.

Unimodal vs. Bimodal Setting: Next, we analyze the effectiveness of our approach in unimodal settings, where only one modality is available during both pseudo-highlight generation and model training. We train the visual unimodal model (V) using visual pseudo-highlights and the audio unimodal model (A) using audio pseudo-highlights. We compare their performance with our approach

in Table 7. While both unimodal settings perform competitively with prior methods, learning from both visual and audio features (i.e., our method) together significantly boosts performance.

Method	TVSum	YouTube	QVHighlights *val*	
	top-5 mAP	mAP	mAP	HIT@1
V(V-PH)	55.02	59.98	16.38	21.35
A(A-PH)	55.45	62.07	17.37	22.06
AV(AV-PH) (Ours)	**60.34**	**68.30**	**18.41**	**26.26**

Table 7: Ablation on unimodal vs. bimodal setting. We analyze the relative contribution of each modality by using only one modality during both pseudo-highlight generation and model training.

Similarity Metrics: To analyze the effectiveness of cosine similarity in our audio and visual pseudo-highlight generation method, we replace it with another popular similarity measure, Pearson's Correlation Coefficient (PCC) [5]. While cosine similarity only considers the similarity of orientation of two feature vectors, PCC considers the linear relationship between features, incorporating both orientation and magnitude. Table 8 indicates that cosine similarity is more effective in capturing the similarity between audio or visual features for pseudo-highlight generation.

Method	TVSum	YouTube	QVHighlights *val*	
	top-5 mAP	mAP	mAP	HIT@1
PCC	52.59	58.71	17.60	23.61
Cosine (Ours)	**60.34**	**68.30**	**18.41**	**26.26**

Table 8: Ablation on similarity metrics in pseudo-highlight generation. We replace the cosine similarity in our method with Pearson's correlation coefficient (PCC).

Audio-Visual Fusion Techniques: In Table 9, we explore various methods of combining audio and visual features during training, as done in previous work [1]. We compare our method with SA^{early} and SA^{late}. In SA^{early}, both audio and visual features are first concatenated and then fed into a single self-attention layer. However, in SA^{late}, each modality is initially processed using separate self-attention layers in a two-stream fashion, and the output features are concatenated for highlight detection. Our fusion scheme, consisting of a bimodal attention module, significantly outperforms these alternative fusion schemes due to its ability to better capture complex cross-modal interactions.

Method	TVSum	YouTube	QVHighlights *val*	
	top-5 mAP	mAP	mAP	HIT@1
SA^{early}	56.68	61.23	16.25	17.87
SA^{late}	53.74	65.61	17.78	23.87
AV(AV-PH) (Ours)	**60.34**	**68.30**	**18.41**	**26.26**

Table 9: Effect of different audio-visual fusion strategies for combining audio and visual features during training.

Comparison with Supervised Setting: In Table 10, we compute the highlight detection performance when real ground-truths are used for training our model instead of the audio-visual pseudo-highlights. This denotes the fully supervised version of our model. Interestingly, our unsupervised method demonstrates strong performance compared to its supervised counterpart. On YouTube, our method underperforms by only 2% compared to the supervised model. This showcases the effectiveness of the proposed pseudo-highlight mechanism as an alternative supervisory signal for highlight detection if manual annotations are not available.

Real Categories vs. Pseudo-categories: To analyze the quality of our generated pseudo-categories for audio-visual pseudo-highlight generation, we replace them with real category information in the datasets. Note that the QVHighlights dataset does not have category information, so we limit

Method	TVSum	YouTube	QVHighlights *val*	
	top-5 mAP	mAP	mAP	HIT@1
AV(Supervised)	68.41	70.18	23.99	32.32
AV(AV-PH) (Ours)	60.34	68.30	18.41	26.26

Table 10: Comparison with the supervised version of our model which uses ground-truth highlight annotations for training instead of our audio-visual pseudo-highlights.

our comparison and analysis to the YouTube and TVSum datasets. Table 11 shows that we obtain better performance using our generated pseudo-categories in comparison to the real categories. We argue that manually annotated category labels can contain ambiguities. Through our clustering-based pseudo-categories, we are able to group semantically related videos that may have been manually labeled as different categories. As a result, our clustering-based pseudo-categories enable us to generate better audio-visual pseudo-highlights compared to using real categories.

Method	TVSum	YouTube
Real categories	56.37	64.28
Pseudo-categories (Ours)	**60.34**	**68.30**

Table 11: Comparison of using pseudo-categories and real categories in our method. Generating audio-visual pseudo-highlights using our clustering-based pseudo-categories performs better.

How does the amount of pseudo-highlights impact the performance? We reduce the amount of training data in each dataset to 25%, 50%, and 75%, yielding different amounts of audio-visual pseudo-highlights. Table 12 shows that training with more pseudo-highlights improves performance. This suggests that using more pseudo-highlights for supervision is beneficial for video highlight detection.

	TVSum	YouTube	QVHighlights *val*	
	top-5 mAP	mAP	mAP	HIT@1
25%	47.31	60.55	17.63	23.23
50%	55.37	62.16	18.13	23.68
75%	54.64	64.25	18.16	24.26
100% (Ours)	**60.34**	**68.30**	**18.41**	**26.26**

Table 12: Impact of the amount of pseudo-highlights on the performance of our model. More pseudo-highlights yield better results.

Impact of different number of clusters, K: In Table 13, we compare the performance of our method for different values of number of clusters, K. Results show a performance drop for non-optimal values of K, while the optimal values chosen by our method in Sec. 2 yield best results.

Number of clusters	6	7	8
TVSum	52.84	54.76	**60.34**
YouTube	**68.30**	63.96	63.56
QVHighlights	17.80	**18.41**	18.02

Table 13: Ablation on the different number of clusters, K. The optimal values of K selected by our method yield best results.

Generalization capability on out-of-distribution data: To assess how our models generalize to out-of-distribution samples, we conducted additional experiments in which we evaluated (see Tables 14 and 15) a model trained on one dataset (YouTube) on another dataset (TVSum), and vice versa.

Despite some drop in performance, the model trained on YouTube outperforms previous methods when evaluated on TVSum. Our TVSum-trained model performs comparably to prior work on YouTube, with the decrease in performance possibly due to the small number of training videos in TVSum. These results suggest that our models have strong generalization capabilities and are able to retain significant efficacy when tested on datasets different from those used in training.

Method	MBF	CVS	SG	DSN	CHD	Ours	Ours (YouTube-trained)
top-5 mAP	34.50	37.20	46.20	42.40	52.76	60.34	55.91

Table 14: Highlight detection results on TVSum. The column, Ours (YouTube-trained), indicates the performance of the model trained on YouTube when evaluated on TVSum.

Method	RRAE	LM-A	LM-S	MN	TC	MT	CHD	Ours	Ours (TVSum-tained)
mAP	38.30	50.50	56.40	61.38	62.97	65.10	65.39	68.30	62.55

Table 15: Highlight detection results on YouTube. The column, Ours (TVSum-trained), indicates the performance of the model trained on TVSum when evaluated on YouTube.

A.5 Additional Qualitative Results

We present additional qualitative results of our method in Figure 4. The video from the YouTube dataset depicts a *dog show*, and the highlights mostly consist of acrobatics, such as jumping over obstacles.

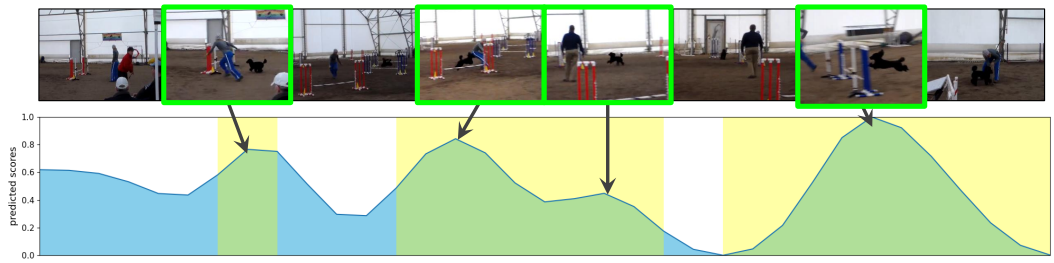

Figure 4: Additional qualitative results. We show the highlight clips (in green) along with the predicted scores of our method (in blue), and the ground truth highlight regions are indicated in yellow. For the *dog show* video from the YouTube dataset, our method correctly selects clips featuring interesting acrobatic movements, such as jumping over obstacles.

Acknowledgements: Zahidul Islam and Mrigank Rochan acknowledge the support of the the University of Saskatchewan and the Natural Sciences and Engineering Research Council of Canada (NSERC).