Test-Time Adaptation for Video Highlight Detection

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

Abstract

Existing video highlight detection methods often struggle to generalize due to varying content, styles, and audio-visual quality in unseen test videos. We propose Highlight-TTA, a test-time adaptation framework for video highlight detection that addresses this limitation by dynamically adapting the model during inference to better align with the specific characteristics of each test video, thereby improving its generalization and highlight detection performance. Highlight-TTA is jointly optimized during training using a self-supervised auxiliary task, *cross-modality hallucinations*, alongside the primary task of highlight detection within a *meta-auxiliary training* scheme to enable effective adaptation. During testing, we adapt the trained model using the self-supervised auxiliary task on the test video to enhance its highlight detection performance. Extensive experiments on three benchmark datasets demonstrate the effectiveness of Highlight-TTA.

1 Introduction

Video highlight detection involves identifying and extracting the most significant and engaging moments from video content [1, 2, 11, 35]. This technology is essential in various fields such as sports, entertainment, education, and social media, where it enhances the user experience by providing quick access to the most compelling segments. However, videos can vary significantly in terms of content, context, and visual/audio quality, making it challenging for existing fixed and generic highlight detection models [28, 35, 32, 1, 2, 19, 17] to perform well across all instances. To tackle this issue, test-time adaptation (TTA) [29] provides a compelling solution. TTA is essential for video highlight detection as it enables the model to adapt to the specific characteristics of each video instance during inference. By adapting the model at test time, it can better account for these variations, leading to more robust highlight detection results tailored to each individual video. We introduce Highlight-TTA, which is, to our knowledge, the first framework to integrate test-time adaptation (TTA) for video highlight detection, to effectively handle diverse video content.

Auxiliary tasks are often used for test-time adaptation (TTA) to provide additional supervision and improve the generalization of the model [29, 4, 9, 10]. By leveraging related but easier-to-solve tasks, the model can learn more robust and transferable representations, which can help improve performance on the primary task, especially when test data differs from training data. In this work, we present a new self-supervised auxiliary task, called cross-modality hallucinations, for video highlight detection. Visual and audio modalities provide complementary information, and recent studies have shown that combining these modalities leads to superior highlight detection performance compared to using a single modality [1, 19]. Our Highlight-TTA framework utilizes the available visual and audio modalities within a video to learn to hallucinate the features of one modality from the other. By performing cross-modality hallucinations, the model can infer missing or obscured information in one modality based on cues from the other, leading to a deeper understanding of the correlation between audio and visual components and thereby enhancing its ability to perform the primary task of highlight detection more accurately. While we could jointly train a model using both auxiliary and primary tasks, recent studies [4, 9] and our experiments suggest that a joint-trained model is prone to bias towards improving the auxiliary task at the expense of the primary task. To address this, we

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

employ a meta-auxiliary learning approach based on model-agnostic meta-learning (MAML) [6] to balance the learning of both tasks and prevent the auxiliary task from overshadowing the primary task. At test time, our model can effectively adapt to each video when fine-tuned using the auxiliary task, ensuring improved highlight detection performance.

In summary, 1) We propose Highlight-TTA, a test-time adaptation approach for video highlight detection, which is to our knowledge, the first work to apply test-time adaptation for highlight detection; 2) We introduce a new self-supervised auxiliary task, cross-modality hallucinations, to improve highlight detection performance; 3) We propose a meta-auxiliary learning scheme to optimize the model parameters, ensuring that adapting these parameters through the auxiliary task during testing improves highlight detection performance; and 4) We conduct extensive experiments on three benchmark datasets, demonstrating the superiority and effectiveness of our approach.

2 Our Approach

Given a video V, we split it into n clips, with each clip containing a fixed number of frames. From each clip c_i (where $i = 1, 2, ...n$), we extract its visual features $v_i \in \mathbb{R}^{d_v}$ from a pre-trained visual feature extractor and audio features $a_i \in \mathbb{R}^{d_a}$ from a pre-trained audio feature extractor. We denote V with its visual features, $\{v_i\}_{i=1}^n$, and audio features, $\{a_i\}_{i=1}^n$. Our method, Highlight-TTA, aims to find a model $F_\theta(V) \rightarrow H$, parameterized by θ that maps V to a set of highlight scores $H = \{h_i\}_{i=1}^n$, where h_i denotes the predicted highlight score of each clip c_i in video V. Our primary loss, L_{pri}, is a binary cross-entropy loss between H and the ground-truth highlight scores, H^{gt}.

Network Architecture: To implement $F_\theta(V)$, we extend two state-of-the-art audio-visual highlight detection networks, JAV [1] and UMT [19], which, at their core, utilize unimodal self-attention modules to encode temporal dependencies within the same modality and bimodal cross-attention modules to capture cross-modal relationships. We extend these networks by introducing two hallucination modules with learnable parameters next to the unimodal self-attention modules for our self-supervised auxiliary task. One hallucination module is fed the self-attended visual features and hallucinates the self-attended audio features, while the other takes in self-attended audio features and learns to hallucinate self-attended visual features. Each of these hallucination modules comprises a self-attention layer sandwiched between two fully-connected layers and includes a skip connection that bypasses the self-attention layer. We denote our model's parameters as $\theta = \{\theta^{shar}, \theta^{pri}, \theta^{aux}\}$. θ^{pri} are the parameters involved only in the primary task, and, θ^{aux} represent the parameters of two cross-modality hallucination modules, which take part only in the auxiliary task. The parameters involved in both the primary and auxiliary tasks are denoted as θ^{shar}.

Cross-Modality Hallucinations: We introduce cross-modality hallucinations as a self-supervised auxiliary task in our framework. This task trains the model to hallucinate one modality from the other (i.e., from visual to audio and vice versa), improving its understanding of audio-visual correlations, enhancing its ability to handle distribution shifts at test time, and ultimately leading to more accurate highlight detection. We compute the cross-modal visual hallucination loss, $L_{a\rightarrow v}^{hal}$, using the MSE loss between the output hallucinated features of our visual hallucination module, $\text{SA}_{a\rightarrow v}^{hal}$, and the self-attended visual features, $\{v_i^v\}_{i=1}^n$. We detach the gradients of $\{v_i^v\}_{i=1}^n$ ensuring backpropagation occurs only through the layers in the audio-modality branch. Similarly, using the audio hallucination module, we calculate the cross-modal audio hallucination loss, $L_{v\rightarrow a}^{hal}$. We add these two losses to compute our auxiliary loss, L_{aux} through which the model learns to minimize the difference between the hallucinated and actual modality features.

Meta-Auxiliary Training: We first train our network using both our primary and auxiliary losses jointly. This joint-trained model is sub-optimal to be directly used for testing as it does not adapt to internal information in test video instances. While we could use the auxiliary task to fine-tune this joint-trained model at test-time, there is a risk that it becomes more biased towards improving the auxiliary task [4, 9, 10, 29] rather than the primary task. Therefore, following prior work [4, 9, 10], we propose a meta-auxiliary training strategy which involves training the model using both auxiliary and primary tasks at training time in a manner that aids adaptation to specific test instance during testing using the auxiliary task. Our meta-auxiliary training algorithm, as shown in Alg. 1, has two loops. In the inner loop, given a video instance V_b and its ground-truth H_b^{gt} from a batch of B videos, along with our joint-trained model parameters θ, we first make a few gradient updates using the auxiliary loss L_{aux}. During these inner loop updates, the model parameters $\{\omega_b^{shar}, \omega_b^{aux}\}$ are

Algorithm 1 Meta-Auxiliary Training for Video Highlight Detection

Input: λ: inner learning rate, γ: meta learning rate, V: video, H^{gt}: ground-truth highlights
Output: θ: learned parameters from meta-auxiliary training
1: Initialize the model with joint-trained weights: $\theta = \{\theta^{shar}, \theta^{aux}, \theta^{pri}\}$
2: **while** not converged **do**
3: Sample a batch of training instances $\{V_b, H_b^{gt}\}_{b=1}^{B}$
4: **for** each b **do**
5: Compute auxiliary cross-modality hallucinations loss: $L_{aux} = L_{a\rightarrow v}^{hal} + L_{v\rightarrow a}^{hal}$
6: Compute adapted parameters: $\{\omega_b^{shar}, \omega_b^{aux}\} \leftarrow \{\theta^{shar}, \theta^{aux}\} - \lambda \nabla_\theta L_{aux}(\{\theta^{shar}, \theta^{aux}\}; V_b)$
7: Auxiliary task update: $\{\theta^{shar}, \theta^{aux}\} \leftarrow \{\theta^{shar}, \theta^{aux}\} - \lambda \nabla_\theta L_{aux}(\{\theta^{shar}, \theta^{aux}\}; V_b)$
8: **end for**
9: Primary task update: $\{\theta^{shar}, \theta^{pri}\} \leftarrow \{\theta^{shar}, \theta^{pri}\} - \gamma \sum_{b=1}^{B} \nabla_\theta L_{pri}(\{\omega_b^{shar}, \theta^{pri}\}; V_b, H_b^{gt})$
10: **end while**

updated as shown in Line 6 of Alg. 1, where λ is the inner learning rate. This inner loop adaptation can be performed during test time since it does not rely on ground-truth highlights. The adapted parameters ω_b^{shar} from the inner updates can then be used to perform our primary task of highlight detection. This step ensures that the performance on the primary task is highly coupled with the updates made in the auxiliary task. This process encourages our model to be updated in such a way that, once adapted to a given video data instance, it enhances the performance of the primary task. As shown in Line 9 of Alg. 1, in the outer update of the algorithm, we optimize our model parameters θ based on the primary loss, where γ is the meta-learning rate.

Test-time Adaptation: At test-time, we use the model obtained from meta-auxiliary training with parameters θ to initialize test-time training. Given a test video V_b, we evaluate the auxiliary loss L_{aux}, which we use to adapt the model to V_b and obtain parameters ω_b^{shar} (Line 6 of Alg. 1) . We use these adapted parameters ω_b^{shar}, along with the parameters θ^{pri}, for highlight detection. We provide detailed information about our approach and the networks used in Appendix A.2.

3 Experiments and Conclusion

We utilize three benchmark video highlight detection datasets, namely YouTube [28], TVSum [27], and QVHighlights [16]. We compare with state-of-the-art methods such as VESD [3], LM [35], Trail. [32], CHD [2], JAV [1] and LSVM [28], on TVSum and YouTube. On QVHighlights, we compare with prior methods without using textual queries: BT [26], CHD [2], JAV [1], and UMT [19]. Additionally, for a strong comparison, we extended CHD, a single-modal method that uses only visual features, to incorporate audio features using early fusion and reported the obtained results. Table 1, 2, and 3 demonstrate the effectiveness of our Highlight-TTA. Note that the results for CHD, JAV, and UMT are from our implementation and runs. The performance of both JAV and UMT improves with the introduction of our Highlight-TTA framework, achieving state-of-the-art results.

	VESD (V)	LM (V)	Trail. (V)	CHD (V)	CHD (AV)	JAV (AV)	**JAV + Ours** (AV)	UMT (AV)	**UMT + Ours** (AV)
t-5 mAP	48.10	56.40	62.80	52.76	55.15	68.42	**70.42**	80.48	**80.96**

Table 1: Highlight detection results on TVSum.

	LSVM (V)	LM (V)	Trail. (V)	CHD (V)	CHD (AV)	JAV (AV)	**JAV + Ours** (AV)	UMT (AV)	**UMT + Ours** (AV)
mAP	53.60	56.40	69.10	65.39	65.72	70.18	**72.26**	74.86	**75.63**

Table 2: Highlight detection results on YouTube.

Ablation Studies: For our ablation studies, we experiment with our Highlight-TTA framework built upon the highlight detection network, JAV [1]. In Table 4, we analyze the impact of various steps in Highlight-TTA and compare against several baseline methods. We first report the results from our

Method	BT (V)	CHD (V)	CHD (AV)	JAV (AV)	**JAV + Ours** (AV)	UMT (AV)	**UMT + Ours** (AV)
mAP	14.36	15.82	17.25	23.98	**24.74**	24.37	**24.47**
HIT@1	20.88	17.10	18.60	29.70	**31.19**	30.97	**31.48**

Table 3: Results on QVHighlights *test* set. We report results obtained from their evaluation server.

Method			TVSum	YouTube	QVHighlights *val*	
Joint	Meta	TTA	t-5 mAP	mAP	mAP	HIT@1
✓			68.30	71.20	24.12	33.16
✓		✓	68.50	71.48	24.13	33.10
✓	✓		70.25	72.03	24.30	33.10
✓	✓	✓	**70.42**	**72.26**	**24.31**	**33.35**

Table 4: Impact of joint-training (Joint), meta-auxiliary learning (Meta), and TTA.

Method	TVSum	YouTube	QVHighlights *val*	
	t-5 mAP	mAP	mAP	HIT@1
JAV + Pseudo-label	66.71	69.84	24.26	32.71
JAV + TENT [31]	69.12	70.36	23.99	32.19
JAV + EATA [21]	69.81	70.67	24.09	32.77
JAV + Ours	**70.42**	**72.26**	**24.31**	**33.35**

Table 5: Comparison of our Highlight-TTA with other test-time adaptation methods.

joint-trained model, which is jointly trained on both the primary loss and the auxiliary cross-modality hallucination loss. Next, we examine the effect of using our auxiliary loss for updating the joint-trained model during test time directly, without the meta-auxiliary training step. It is noteworthy that this simple method already outperforms existing methods, demonstrating the effectiveness of using cross-modality hallucinations as an auxiliary task for highlight detection. Finally, in the last row, our Highlight-TTA which integrates joint-training, meta-auxiliary training, and test-time adaptation outperforms all of the baseline methods. Moreover, to evaluate the effectiveness of Highlight-TTA, in Table 5, we compare it with popular TTA methods, including TENT [31] and EATA [21]. We also compare it with a simple confidence-based pseudo-labeling technique [15, 33, 34, 20], using this technique as an auxiliary task instead of the cross-modality hallucinations in our Highlight-TTA. We present additional experiment details and ablation studies in Appendix A.3 and A.4, respectively.

Qualitative Results: In Fig. 1, we compare the predictions of a generic, fixed joint-trained model based on JAV with those of the JAV + Highlight-TTA (ours) on a test video from TVSum, which achieves better alignment with the ground-truth highlights.

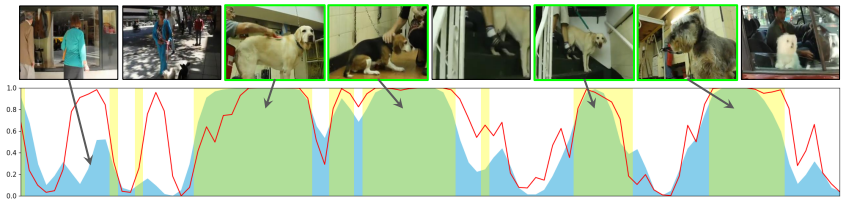

Figure 1: Qualitative results on an example test video from TVSum. The yellow region corresponds to the ground-truth annotations. We indicate the predicted scores of the joint-trained JAV model with a red line, while the sky-blue region represents the predictions of JAV + Highlight-TTA (ours).

Conclusion: Recent highlight detection methods, despite their advancements, face significant challenges in generalizing to unseen test videos. These methods typically rely on a generic highlight detection model for each test video, which does not account for the unique characteristics and variations of individual test videos. Consequently, their performance suffers during testing. To address this, we propose Highlight-TTA, a test-time adaptation framework for video highlight detection. Our approach involves jointly optimizing Highlight-TTA using a self-supervised auxiliary task, cross-modality hallucinations, alongside the primary task of highlight detection in a meta-auxiliary training scheme to enable effective adaptation during test-time. Extensive experiments and ablation studies conducted on three benchmark datasets demonstrate the effectiveness of Highlight-TTA.

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*, pages 8127–8137, 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*, pages 14042–14052, 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 (ECCV)*, pages 184–200, 2018.

[4] Zhixiang Chi, Yang Wang, Yuanhao Yu, and Jin Tang. Test-time fast adaptation for dynamic scene deblurring via meta-auxiliary learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9137–9146, 2021.

[5] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019.

[6] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[7] 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 *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 776–780. IEEE, 2017.

[8] Michael Gygli, Yale Song, and Liangliang Cao. Video2gif: Automatic generation of animated gifs from video. In *IEEE conference on computer vision and pattern recognition*, pages 1001–1009, 2016.

[9] Ahmed Hatem, Yiming Qian, and Yang Wang. Point-tta: Test-time adaptation for point cloud registration using multitask meta-auxiliary learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16494–16504, 2023.

[10] Ahmed Hatem, Yiming Qian, and Yang Wang. Test-time adaptation for point cloud upsampling using meta-learning. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1284–1291. IEEE, 2023.

[11] Fa-Ting Hong, Xuanteng Huang, Wei-Hong Li, and Wei-Shi Zheng. Mini-net: Multiple instance ranking network for video highlight detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 345–360. Springer, 2020.

[12] 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. Springer, 2018.

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

[14] 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.

[15] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013.

[16] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. *Advances in Neural Information Processing Systems*, 34:11846–11858, 2021.

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

[18] Shikun Liu, Andrew Davison, and Edward Johns. Self-supervised generalisation with meta auxiliary learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[19] 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*, pages 3042–3051, 2022.

[20] Islam Nassar, Munawar Hayat, Ehsan Abbasnejad, Hamid Rezatofighi, and Gholamreza Haffari. Protocon: Pseudo-label refinement via online clustering and prototypical consistency for efficient semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11641–11650, 2023.

[21] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *The Internetional Conference on Machine Learning*, 2022.

[22] 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*, pages 3657–3666, 2017.

[23] 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 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[24] 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*, pages 8748–8763. PMLR, 2021.

[25] Mrigank Rochan and Yang Wang. Video summarization by learning from unpaired data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7902–7911, 2019.

[26] Yale Song, Miriam Redi, Jordi Vallmitjana, and Alejandro Jaimes. To click or not to click: Automatic selection of beautiful thumbnails from videos. In *25th ACM international on conference on information and knowledge management*, pages 659–668, 2016.

[27] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. In *IEEE conference on computer vision and pattern recognition*, pages 5179–5187, 2015.

[28] Min Sun, Ali Farhadi, and Steve Seitz. Ranking domain-specific highlights by analyzing edited videos. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pages 787–802. Springer, 2014.

[29] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pages 9229–9248. PMLR, 2020.

[30] 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 I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[31] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

[32] Lezi Wang, Dong Liu, Rohit Puri, and Dimitris N Metaxas. Learning trailer moments in full-length movies with co-contrastive attention. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 300–316. Springer, 2020.

[33] Xudong Wang, Zhirong Wu, Long Lian, and Stella X Yu. Debiased learning from naturally imbalanced pseudo-labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14647–14657, 2022.

[34] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 33:6256–6268, 2020.

[35] 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*, pages 1258–1267, 2019.

[36] 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*, pages 4633–4641, 2015.

[37] 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*, pages 7950–7959, 2021.

[38] 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*, volume 32, 2018.

A Appendix

A.1 Related Work

Video highlight detection has garnered significant attention in recent years, driven by the growing demand for efficient content viewing and summarization. Many approaches have been proposed to tackle this problem, spanning from traditional rule-based techniques to cutting-edge deep learning based methods. Early works often relied on handcrafted features and heuristics to identify key moments or segments within videos [27]. However, with the advent of deep learning, researchers have shifted towards data-driven approaches for highlight detection that learn representations directly from raw video data. Many of high performing highlight detection approaches rely on manually annotated frame-level supervision [8, 12, 32, 38, 1, 19]. To alleviate the burden of acquiring costly labeled data, some methods leverage cheaper video-level tags or category information for weak supervision [36, 3, 11, 22, 35, 37]. However, these approaches often require access to large-scale external datasets for training. Notably, some recent unsupervised methods have also shown promising results [2, 17]. Despite these advancements, existing methods typically struggle to generalize well since they focus on a generic highlight detection model, which suffers when applied directly to new, unseen testing videos due to distribution shifts between training and testing data. To address this challenge, inspired by recent success of test-time adaptation [29], we propose Highlight-TTA, a test-time adaptation framework for video highlight detection.

Our work is related to a popular meta-learning algorithm, MAML [6], which enables rapid learning and adaptation to a new task using only a few training examples. Additionally, our work is connected to the meta-auxiliary learning framework (MAXL) [18], which generates auxiliary labels to enhance the primary task using ground-truth labels. Recent studies in point clouds [9, 10] and image deblurring [4] have explored meta-auxiliary learning combined with test-time adaptation. To our knowledge, we are the first to explore test-time adaptation using meta-auxiliary learning for video highlight detection. Additionally, we introduce a new self-supervised auxiliary task, cross-modal hallucinations, to leverage multimodal audio-visual information for effective video highlight detection.

A.2 Additional Details of Our Approach

We illustrate our Highlight-TTA framework in Figure 2 where we lay out our meta-auxiliary training and test-time adaptation method for video highlight detection. Our meta-auxiliary training mechanism for enabling effective and fast adaptation to each test instance utilizes two loop. We first obtain adapted parameters using the auxiliary loss present in the inner loop. Next, we evaluate and update the model weights in the outer loop by using the primary loss computed from the adapted parameters. Given a test video instance during testing, we update the parameters of model using the auxiliary task to adapt specifically to the given test video. With the adapted model, we detect highlights of the given video.

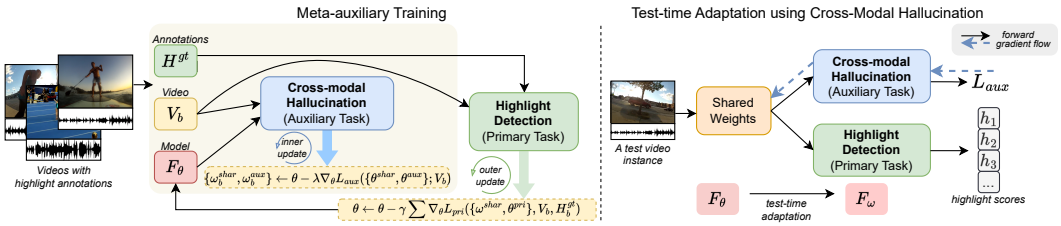

Figure 2: Illustration of our meta-auxiliary training and test-time adaptation method for video highlight detection. During the meta-auxiliary training stage, we initially obtain adapted parameters using the auxiliary loss present in the inner loop. Next, we evaluate and update the model weights in the outer loop by using the primary loss computed from the adapted parameters. At test-time, we update the model using the auxiliary task to adapt specifically to a given test video instance.

Next, we describe the architecture of our highlight detection network built upon JAV [1] in more detail here. Fig. 3 illustrates a schematic overview of our network. At its core, our network initially employs unimodal self-attention layers [30] 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. We built upon this network by introducing two cross-modal hallucination modules, one to hallucinate self-attended audio-features while the other hallucinates self-attended visual features.

Concretely, given a video with n clips, our audio-visual model firstly processes the clip-level visual features $\{v_i\}_{i=1}^n$ using a self-attention layer $\text{SA}_{v\to v}$ and the clip-level audio features $\{a_i\}_{i=1}^n$ using another self-attention layer $\text{SA}_{a\to a}$. Then, we feed the self-attended visual features $\{v_i^v\}_{i=1}^n$ into our cross-modal audio hallucination module $\text{SA}_{v\to a}^{hal}$ to hallucinate self-attended audio features $\{a_i^a\}_{i=1}^n$. Similarly, we send the self-attended audio features $\{a_i^a\}_{i=1}^n$ to our cross-modality visual hallucination module $\text{SA}_{a\to v}^{hal}$, to hallucinate the self-attended visual features $\{v_i^v\}_{i=1}^n$. On the other hand, two bimodal attention layers $\text{BMA}_{v\to a}$ and $\text{BMA}_{a\to v}$ are also fed 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 the self-attended and bimodal attended features using a set of learnable weights and passes them through two fully-connected layers to predict the highlight score h_i for each clip in the video V.

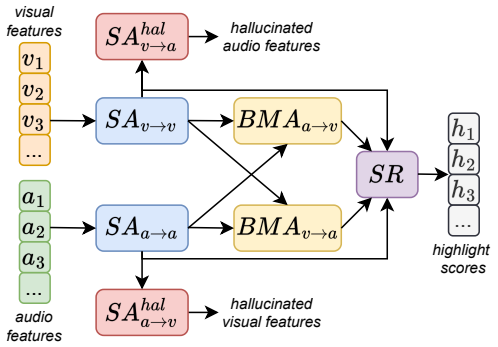

Figure 3: Illustration of our network architecture.

As mentioned in Sec. 2, we also incorporate our Highlight-TTA framework into another state-of-the-art audio-visual highlight detection network, UMT [19]. UMT is a multi-modal transformer model that includes two unimodal encoders, one for video and one for audio. The features from these unimodal encoders are fused using a cross-modal encoder to capture cross-modal dependencies. Similar to JAV, we extend UMT with two cross-modal hallucination modules by reusing its unimodal encoder modules.

A.3 Datasets and Settings

We utilize three benchmark video highlight detection datasets, namely YouTube [28], TVSum [27], and QVHighlights [16]. YouTube contains videos from six classes- parkour, dog, gymnastics, skating, skiing, and surfing with about 100 videos in each category. We utilize the standard train-test splits provided with the dataset. TVSum is a smaller dataset with 50 videos across 10 categories including changing vehicle tire, making sandwich, and so forth. We follow prior works [1, 2, 25] and utilize a random train-test split with a ratio of 80:20. We run our experiments on TVSum five times and report the average performance. QVHighlights is a large dataset containing about 10,000 videos. The dataset is primarily designed for query-focused video highlight detection and moment retrieval. Each video is associated with corresponding textual queries and saliency/highlight scores. The dataset has canonical 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, we evaluate our method against prior non-query-based methods on this dataset.

Features: Following prior work on TVSum and YouTube [2, 1], we extract visual features from each clip using a 3D-CNN (ResNet-34) backbone. For the videos of QVHighlights, following [16, 19], we extract visual features using SlowFast [5] and video encoder of CLIP (ViT-B/32) [24]. To extract audio features, on all three datasets, we use PANN [14] audio network pre-trained on AudioSet [7].

Evaluation Metrics: On QVHighlights, we utilize Mean Average Precision (mAP) for evaluation, which takes into account the highlight scores of all the clips, and HIT@1, which considers the hit ratio of the clip with the highest score for each video. Following prior work [16], we consider only the clips rated as *Very Good* by users to be highlights during evaluation. Following existing works [2, 1], on YouTube, we report Mean Average Precision (mAP), and on TVSum, we report the mean average precision on the top five predicted clips (top-5 mAP). We report all metrics as percentages.

Implementation Details: We implement our method using PyTorch [23] and use the official implementations of JAV [1] and UMT [19] as the backbones in our method. In the outer loop update of meta-auxiliary training (Algorithm 1), we use the Adam optimizer [13] with a learning rate of $\lambda = 5 \times 10^{-5}$. We use the same optimizer and learning rate for joint-training as well. For the inner update of our meta-auxiliary training and during test-time adaptation, we use the SGD optimizer with a learning rate of $\gamma = 1 \times 10^{-1}$. We employ three gradient updates during training and testing to adapt our model using the auxiliary loss in Line 6 of Alg. 1. We implement our Highlight-TTA framework by building upon JAV [1] and UMT [19]. For the JAV model, on the YouTube dataset, we train for 30 epochs during joint training and 10 epochs during meta-auxiliary learning. On TVSum, we train for 100 epochs during joint training and 20 epochs for meta-auxiliary learning. On the larger QVHighlights dataset, we train for 15 epochs in both phases. For the UMT model, we train for 100 epochs on the YouTube dataset during both joint training and meta-auxiliary learning. On the TVSum dataset, we train for 200 epochs in both phases, and on the QVHighlights dataset, we train for 50 epochs during both phases. We train our models on one NVIDIA GeForce RTX 2080 Ti 12GB GPU.

A.4 Additional Ablation Experiments

Number of gradient updates: In Table 6, we examine the impact of the number of gradient updates using the cross-modality hallucinations auxiliary task in the inner loop of meta-auxiliary training (Alg. 1) and test-time adaptation of Highlight-TTA (Sec. 2). We use the same number of gradient updates in both training and testing, which is intuitive and has been found useful in prior work [4, 10]. Overall, we find that our model performs best with three gradient updates across all datasets, allowing the model to sufficiently adapt to the internal information of each test instance. However, further increasing the number of updates does not yield any additional performance boost.

No. of gradient updates	TVSum	YouTube	QVHighlights *val*	
	top-5 mAP	mAP	mAP	HIT@1
1	68.68	71.04	24.26	31.74
2	69.08	71.07	24.07	32.13
3	**70.42**	**72.26**	**24.31**	**33.35**

Table 6: Impact of number of gradient updates on model performance using cross-modality hallucinations for meta-auxiliary training and test-time adaptation.

A.5 Additional Qualitative Results

In Fig. 4, we present additional qualitative results on a test video on *surfing* from the YouTube dataset. We again compare the predictions of a generic, fixed joint-trained model based on JAV with those of JAV + Highlight-TTA (ours), which improves highlight detection predictions and aligns better with the ground-truth highlights.

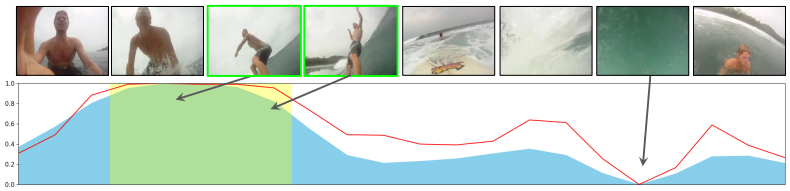

Figure 4: Additional qualitative results. We denote the ground-truth highlight annotations as yellow regions. The predicted scores of the joint-trained model are indicated by a red line, while the sky-blue region represents the predictions of our Highlight-TTA. For the *surfing* video from the YouTube dataset, the overall alignment of highlight predictions with ground-truth highlights improves with Highlight-TTA, as the highlight score for the non-highlight clips decreases.

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).