A APPENDIX

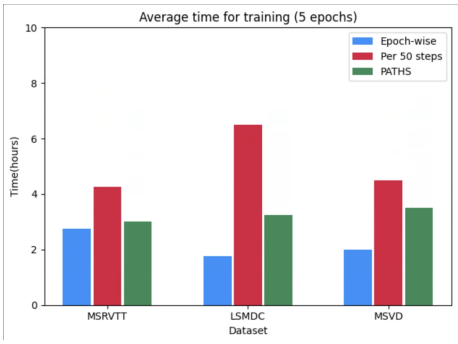

Figure 5: The full training time in hours are compared. For training, each evaluation scheme has been adopted. Each bar is an average time of all models used in the experimentations.

Methods	Accuracy↑
ClipBERT (Lei et al., 2021)	37.4
VGT (Xiao et al., 2022)	39.7
VQA-T (Yang et al., 2021)	41.5
SiaSamRea (Yu et al., 2021)	41.6
MERLOT (Zellers et al., 2021)	43.1
Co-Tok (Piergiovanni et al., 2022)	45.7
*EMCL (Jin et al., 2022)	45.2
*EMCL (+PATHS)	**45.5(+0.3)**

Table 5: Performance comparison of video question answering on MSRVTT. PATHS can also be applied to EMCL for the task, which achieves performance improvement. We use original code of EMCL for the experimental evaluation.

A.1 TIME EFFICIENCY

In Figure 5, we present a comparative analysis of time complexities across three evaluation paradigms: traditional epoch-wise evaluation, N-step evaluation, and the PATHS evaluation method. In text-to-video retrieval literature, all models are trained over 5 epochs, and we follow the protocol reporting the time for training over 5 epochs. Figure 5 is the average total training time in hour of the previous models reported in Table 2, and 3. The number of videos and, the number of frames of each video have strong differences across three datasets, which is reflected in the bar graph in Figure 5.

Notably, N-step evaluation requires significant increase in time for training each of the datasets. This is mainly because of the frequent evaluations on validation set at every epoch, which causes computational overhead. While many recent studies take the N-step approach, less attention has been paid on the computational overhead problem. Our study addresses this problem proposing a novel approach which doesn't require frequent evaluations. For MSRVTT dataset, PATHS only takes approximately the same amount of time for training as the conventional epoch-wise method. For both LSMDC and MSVD datasets, PATH saves the training time compared to the 50-step approach. When number of video gets large, such as the LSMDC dataset, the improvement becomes significant. It is also worth to note that PATHS performs as nearly as the N-step evaluation when compared one-to-one (see per 50 step vs first stage in Table 4).

A.2 PATHS APPLIED TO VIDEO QUESTION ANSWERING TASK

We further investigate how PATHS can be applied to a task other than text-to-video retrieval. Several works have been proposed in video question answering (VideoQA) task. VideoQA leverages visual information from videos to predict corresponding answers of input questions. Utilizing the target vocabulary designed for the MSRVTT-QA dataset (Xu et al., 2016), we trained a fully connected layer atop the final linguistic features to categorize the answer.

In Table 5, EMCL (Jin et al., 2022) is the only model that leverages the pre-trained weights from CLIP. We use EMCL as a baseline for the given task, and see how PATH can further improve EMCL in VideoQA. From Table 5, we confirm that PATHS can also improves EMCL on VideoQA task, which highlights its generalization to other tasks.

A.3 THE PROCESS OF THE STMA

This section elaborates the mechanics STMA, which is conceptualized in Figure 6. The STMA begins by bisecting a video based on a scene transition algorithm, encapsulated within the green

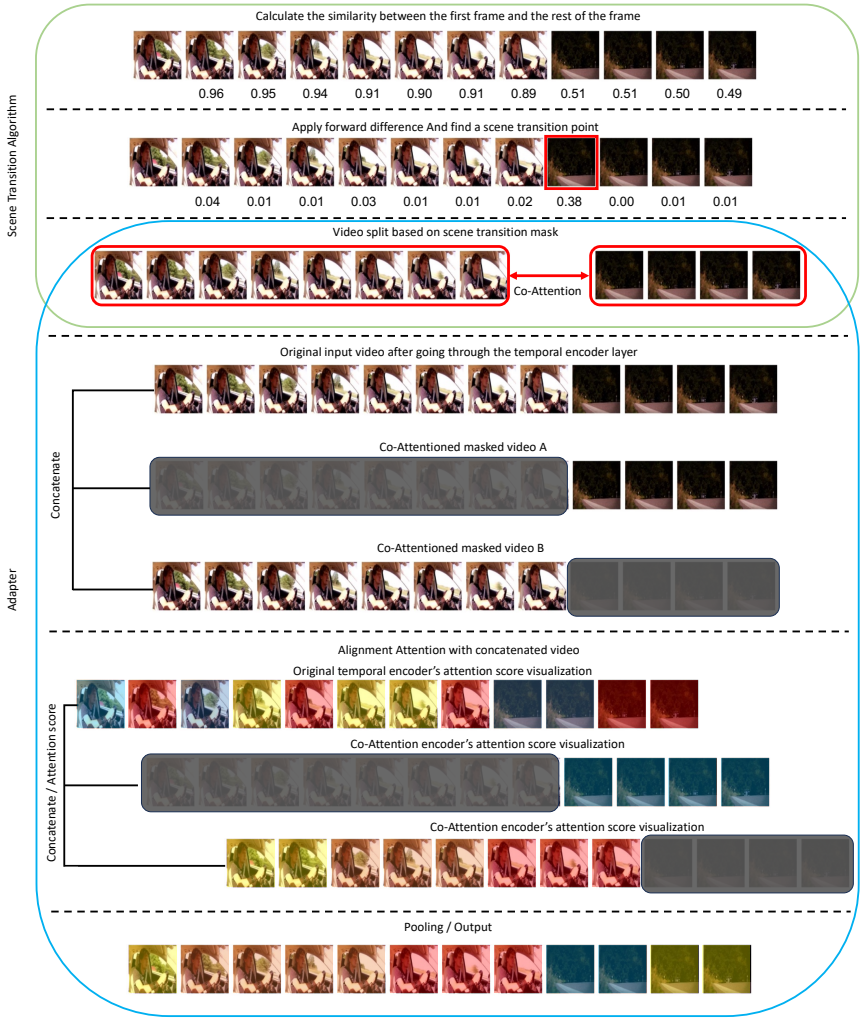

Figure 6: The figure presented serves as a schematic representation of our STMA. The segment enclosed within the green box illustrates the algorithm's mechanism for identifying scene transition points. The blue box emphasizes the functional deployment of STMA within the temporal encoder. The color presents the attention score: red signifies high attention scores, yellow indicates medium levels, and blue represents low scores.

box in the figure. This algorithm initiates by assessing the similarity between the initial frame and subsequent frames as detailed in Section 4.2. We then identify the scene transition point by the similarity matrix. After that, we create a mask before and after the corresponding scene transition point.

The blue box in the figure outlines the process of STMA. This module ingests two masks generated from the scene transition algorithm, and processes three frame sequences within the model. The first sequence is the original video post-attention layer, while the remaining sequences are mask-modified frames subjected to co-attention mechanisms. The multiple processes allows the model to capture the multiple aspects of scenes within a given video. Finally, we apply additional attention layer and pooling layer for obtaining final output which reflects the importance among the three processes.

Our model internalizes knowledge of scenes originating from a singular video source through the co-attention module. We use a single encoder so that the alignment attention module and co-attention

module share the same encoder. This approach can minimize the additional parameters, and also turn out to be more effective in performance than having two separate encoder. In general, training co-attention and multi-head attention simultaneously in the same encoder hinders effective optimizations. However, in this unique scenario, where each of the frames from a same video exhibit strong correlations, the problem is mitigated. We can additionally mitigate the problem with our training strategy. In the second stage of our training scheme, all irrelevant parameters are kept frozen, and thus attention weights can be updated effectively.

A.4 Ablation Study for STMA

Gate	Text-to-Video Retrieval			
	R@1	R@5	R@10	MeanR
Baseline	47.2	73.5	82.5	13.8
w/o CoAttn	47.8	73.3	81.8	13.6
w/o A-Attn	47.8	73.4	82.2	13.7
Full STMA	**48.4**	**73.7**	**82.7**	**13.2**

Table 6: Ablation study of attention mechanisms on STMA. Text-to-video retrieval results on MSRVTT are compared. We use X-CLIP as a baseline.

Gate	Text-to-Video Retrieval			
	R@1	R@5	R@10	MeanR
Baseline	47.2	73.5	82.5	13.8
Avg	48.1	73.6	**82.9**	13.4
Max	48.1	73.4	**82.9**	**13.2**
Conv	**48.4**	**73.7**	82.7	**13.2**

Table 7: Ablation study of pooling strategies on STMA. Text-to-video retrieval results on MSRVTT are compared. We use X-CLIP as a baseline.

To understand how the co-attention and alignment attention contribute to the STMA performance, we perform ablative analysis, where the results are provided in Table 6. We confirm that each component indeed contributes to the STMA performance. The full implementation of STMA brings significant performance improvements. More importantly, in full implementation, each component contributes complementarily to the final results.

Additionally, Table 7 illustrates the performance with different pooling techniques employed in the STMA. We compare average pooling, max pooling, and convolution-1D pooling methods. Among these, the convolution-1D approach demonstrated the best results.

A.5 Video-to-Text Retrieval

Methods	MSRVTT							
	R@1↑	R@5↑	R@10↑	MeanR↓	R@1↑	R@5↑	R@10↑	MeanR↓
	ViT-B-32				ViT-B-16			
CenterCLIP (Zhao et al., 2022) SIGIR'22	45.1	72.4	83.1	10.0	47.7	75.0	83.3	10.2
DRL (Wang et al., 2022) WSDM'22	45.3	73.9	83.3	-	48.9	76.3	85.4	-
*X-Pool (Gorti et al., 2022) CVPR'22	44.4	73.3	84.0	9.0	-	-	-	-
MEME (Kang & Cho, 2023) SIGIR'23	47.7	74.0	83.3	9.4	-	-	-	-
*CLIP4Clip (Luo et al., 2022) Neurocomputing'22	43.2	70.5	80.2	11.8	45.7	72.4	**83.2**	10.8
*CLIP4Clip (+PATHS)	**43.4(+0.2)**	**71.5(+1.0)**	**81.4(+1.2)**	**10.8(+1.0)**	**46.3(+0.6)**	**75.3(+2.9)**	83.0(-0.2)	**9.6(-1.2)**
*TS2-Net (Liu et al., 2022) ECCV'22	**46.1**	73.8	83.5	9.4	46.8	76.7	84.8	8.6
*TS2-Net (+PATHS)	45.9(-0.2)	**74.0(+0.2)**	**84.4(+0.9)**	**8.9(-0.5)**	**47.9(+1.1)**	**77.4(+0.7)**	**86.6(+1.8)**	**8.2(-0.4))**
*EMCL-Net (Jin et al., 2022) NeurIPS'22	46.1	73.7	**84.2**	9.8	50.2	75.7	84.0	8.8
*EMCL-Net (+PATHS)	**48.0(+1.9)**	**74.8(+1.1)**	83.8(-0.4)	**9.2(-0.6)**	**51.1(+0.9)**	**77.0(+1.3)**	**85.1(+1.1)**	**8.3(-0.5)**
*X-CLIP (Ma et al., 2022) ACMMM'22	47.2	73.2	80.6	10.5	47.6	**77.3**	84.8	8.8
*X-CLIP (+PATHS)	**47.7(+0.5)**	**73.6(+0.4)**	**82.3(+1.7)**	**9.5(-1.0)**	**48.8(+1.2)**	75.9(-1.4)	**84.9(+0.1)**	**8.6(-0.2)**
*DiCoSA (Jin et al., 2023b) IJCAI'23	**47.6**	74.2	83.7	8.8	49.9	**77.8**	85.3	8.5
*DiCoSA (+PATHS)	47.2(-0.4)	**74.9(+0.7)**	**83.9(+0.2)**	**8.7(-0.1)**	**50.2(+0.3)**	77.1(-0.6)	**86.7(+1.4)**	**7.9(-0.6)**

Table 8: Video-to-text retrieval task results on MSRVTT, $*$ denotes that the results are reproduced using the publicly released code.

Following previous studies in text-to-video retrieval, we perform video-to-text retrieval task. We report the performance in Table 8, where we use MSRVTT dataset for the evaluation.The experimental setup follows the setups from the existing backbone models. We confirm that PATHS is always effective when applied to any strong baseline models for video-to-text retrieval tasks.

A.6 Qualitative Analysis

To gain insights into the attention weights assigned by STMA and to evaluate its efficacy in scene recognition, we perform qualitative analysis. In the associated figures, attention weights are ex-

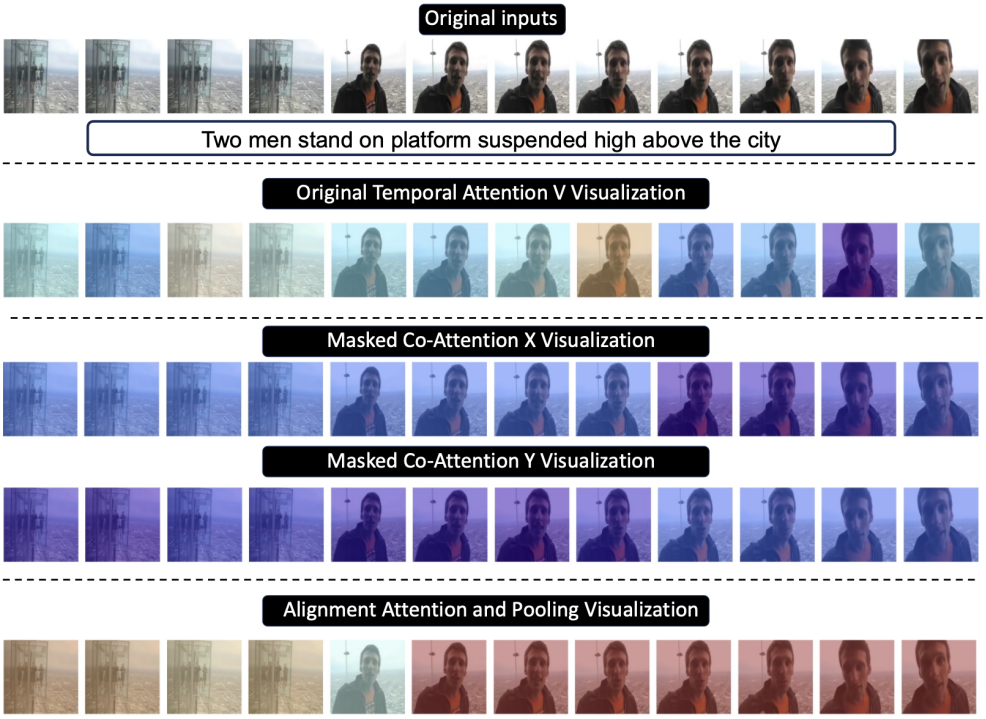

Figure 7: Visualization of the attention score extracted at each part of SMTA.

pressed with colors; red signifies high attention, yellow indicates medium attention, and blue represents low attention. The images in Figure 7, 8, 9, and 10 depict actual experimental results. Distinct from the visualizations generated by original temporal attention, the final output from STMA clearly identifies a sequence of scenes around the transition point. This substantiates that our model is proficient in contextual scene learning.

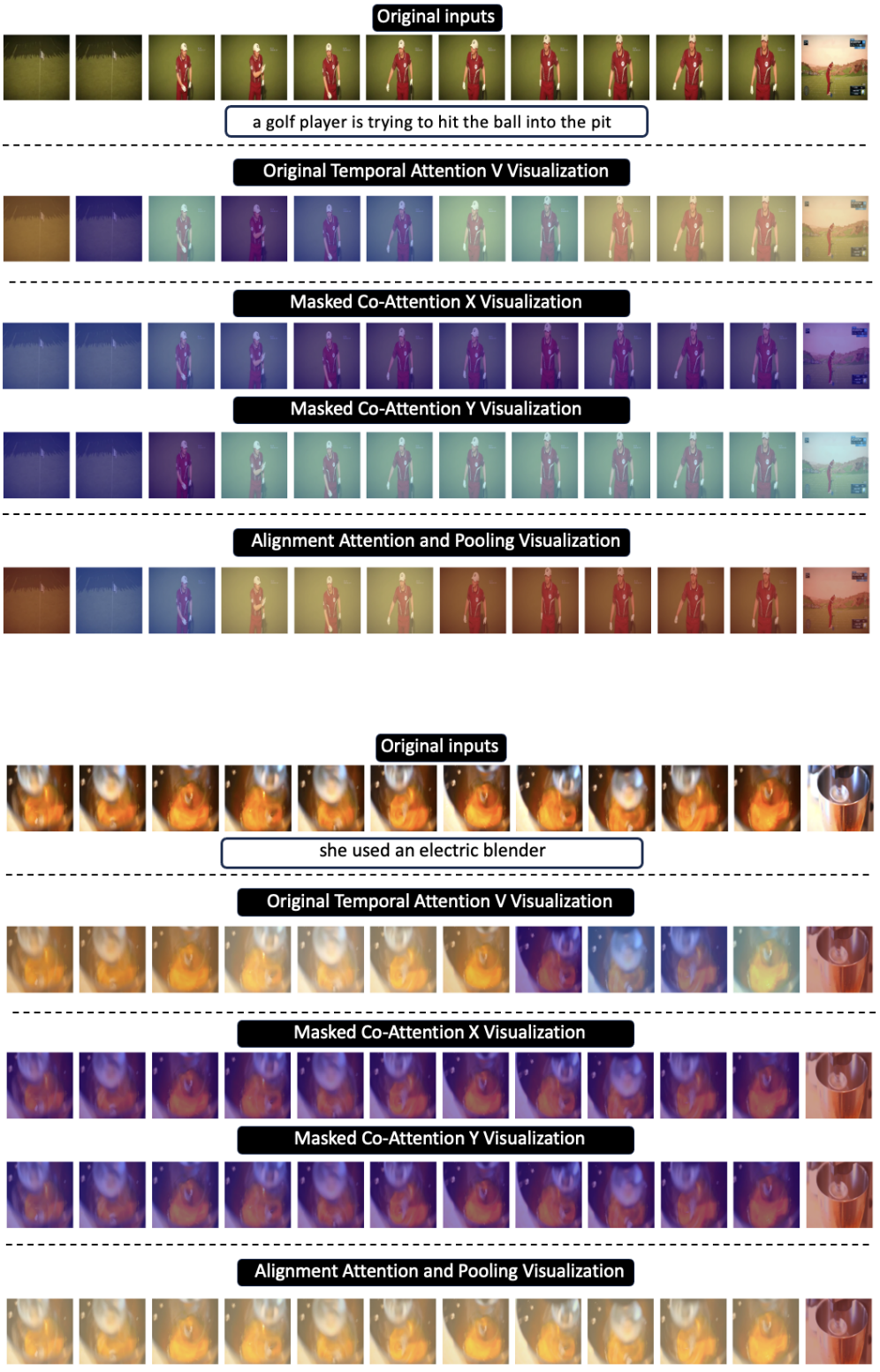

Figure 8: Visualization of the attention score extracted at each part of SMTA.

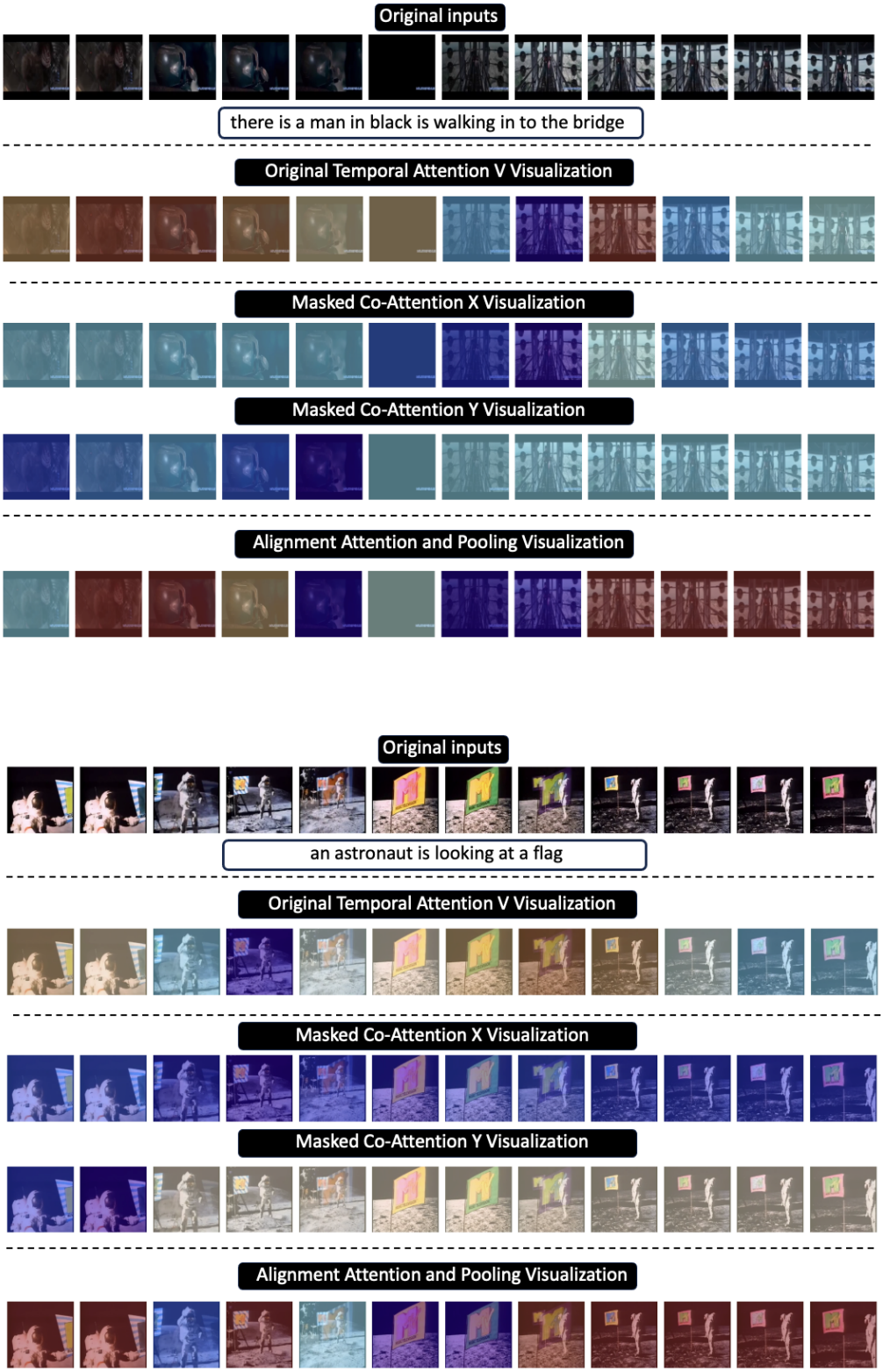

Figure 9: Visualization of the attention score extracted at each part of SMTA.

Figure 10: Visualization of the attention score extracted at each part of SMTA.