Supplementary Material for Progressive Attention on Multi-Level Dense Difference Maps for Generic Event Boundary Detection

Jiaqi Tang[1] Zhaoyang Liu[2] Chen Qian[2] Wayne Wu[2] Limin Wang[1] ✉
[1]State Key Laboratory for Novel Software Technology, Nanjing University, China
[2]SenseTime Research
jqtang@smail.nju.edu.cn, zyliumy@gmail.com, {qianchen, wuwenyan}@sensetime.com, lmwang@nju.edu.cn

A. Additional Ablation Study

Study on Multi-Level Spatial Architecture. To further analyze the design of our multi-level spatial feature bank, we perform comparisons of different multi-level spatial architectures in Table A. To be specific, we respectively employ feature pyramid network (FPN [8]) and pyramidal feature hierarchy (SSD [10]) as the multi-level spatial feature extractor of DDM-Net, and compare the performance. Different from prior knowledge in object detection, DDM-Net witnesses an inferior performance when combined with FPN. We analyze that lateral and top-down connection layers of FPN are trained without explicit spatial location supervisions (*e.g.*, bounding boxes) in GEBD task [12], thus leading to insufficient training of those layers and overall performance degradation.

Spatial architecture	0.05	0.25	0.5	Average
Feature pyramid network	0.7511	0.8697	0.8815	0.8557
Pyramidal feature hierarchy	**0.7643**	**0.8870**	**0.9016**	**0.8726**

Table A. Study on multi-level spatial architecture on Kinetics-GEBD, measured by F1 score at different Rel.Dis. thresholds.

Study on the Number of Attention Layers. We experiment on the number of attention layers of intra-modal attention module and cross-modal attention module, and display the results in Table B. DDM-Net achieves the best performance with 6 intra-modal attention layers and 6 cross-modal attention layers. With the increase of the number of attention layers, the performance gain of increasing layers decreases.

Study on the Number of Learnable Queries ω. The number of learnable queries ω influences the performance of DDM-Net, as demonstrated in Table C. Too few queries ($\omega = 1$) are not enough to capture all patterns, while too many queries ($\omega = 10$) lead to redundant intra-modal features. In experiments, we observe that DDM-Net reaches the best performance when ω is set to 5.

Intra	Cross	0.05	0.25	0.5	Average
1	1	0.7584	0.8774	0.8895	0.8633
3	3	0.7622	0.8841	0.8983	0.8698
6	6	**0.7643**	**0.8870**	**0.9016**	**0.8726**

Table B. Study on the number of attention layers on Kinetics-GEBD, measured by F1 score at different Rel.Dis. thresholds.

ω	0.05	0.25	0.5	Average
1	0.7579	0.8763	0.8884	0.8622
3	0.7614	0.8853	0.9004	0.8709
5	**0.7643**	**0.8870**	**0.9016**	**0.8726**
10	0.7592	0.8765	0.8888	0.8626

Table C. Study on the number of learnable queries ω on Kinetics-GEBD, measured by F1 score at different Rel.Dis. thresholds.

Study on Positional Embeddings of Cross-Modal Attention. We conduct ablations on positional embeddings of cross-modal attention module, as shown in Table D. Adding positional embeddings in the cross-modal attention module harms the performance, hence we analyze that cross-modality feature aggregation should be directly operated on raw features. It is worth noting that learnable positional embeddings of intra-modal attention module cannot be removed, as they are employed to localize key features of clips, similar to positional embeddings in previous detection tasks (*e.g.*, to localize objects in object detection, to localize actions in temporal action detection).

Study on Balanced Sampler. Since boundaries and non-boundaries are extremely imbalanced (about 1:6), we follow [12] to exploit the same balanced sampler. As shown in Table E, the performance of balanced sampler is better.

✉: Corresponding author.

Aggregation	0.05	0.25	0.5	Average
cross w/ PE	0.7510	0.8720	0.8841	0.8576
cross w/o PE	0.7590	0.8770	0.8894	0.8631
intra + cross w/ PE	0.7597	0.8801	0.8931	0.8659
intra + cross w/o PE	**0.7643**	**0.8870**	**0.9016**	**0.8726**

* PE: positional embedding, w/: with, w/o: without.

Table D. Study on positional embedding of cross-modal attention module on Kinetics-GEBD, measured by F1 score at different Rel.Dis. thresholds.

Sampler	0.05	0.25	0.5	Average
Plain sampler	0.7456	0.8784	0.8932	0.8627
Balanced sampler	**0.7643**	**0.8870**	**0.9016**	**0.8726**

Table E. Study on balanced sampler on Kinetics-GEBD, measured by F1 score at different Rel.Dis. thresholds.

B. More Results

More comparisons of different representations. Due to the limited space, we present the best performance of each representation in Table 4a of the main paper. In this section, we display the complete result of each representation and perform more comparisons of different representations. *First,* only DDM obtains the best performance when stride s is set to 6, indicating that compared with other representations, DDM can take advantage of larger temporal contexts. *Second*, if we fairly compare the representations under the same setting $s = 3$, DDM still outperforms other representations, demonstrating the effectiveness of dense motion representation in GEBD task. *Third*, in Table G, pairwise flow and RGB differences are superior to consecutive (non-pairwise) flow and RGB differences at the most strict threshold (Rel.Dis. = 0.05), demonstrating the effectiveness of pairwise calculation in GEBD task.

Representation	w	s	0.05	0.25	0.5	Average
RGB	5	3	**0.6793**	**0.8589**	**0.8772**	**0.8375**
RGB	5	6	0.6118	0.8462	0.8772	0.8180
Flow	5	3	**0.6625**	**0.8045**	**0.8206**	**0.7877**
Flow	5	6	0.6091	0.7703	0.7975	0.7530
RGB diff	5	3	**0.7272**	**0.8591**	**0.8753**	**0.8440**
RGB diff	5	6	0.6638	0.8629	0.8876	0.8399
DDM	5	3	0.7476	0.8688	0.8813	0.8544
DDM	5	6	**0.7512**	**0.8738**	**0.8861**	**0.8591**

* w and s are defined in the main paper.

Table F. More comparisons of different representations on Kinetics-GEBD, measured by F1 score at different Rel.Dis. thresholds.

DDM-Net with CSN backbone. Owing to the limited space of the main paper, we report the performance of DDM-Net on testing set when it is combined with IG-65M [3] pretrained CSN [14] backbone network. To validate our method, we further perform ablations on the validation set (annotations of the testing set are not available, entries to the testing server are limited). In Table H, we observe that DDM-Net can still increase the performance of powerful CSN representations by nearly 2 percent, from 79.3% to 81.3%.

Representation	0.05	0.25	0.5	Average
Flow	0.6625	**0.8045**	**0.8206**	**0.7877**
Pairwise Flow	**0.7012**	0.7910	0.7998	0.7806
RGB diff	0.7272	0.8591	0.8753	0.8440
Pairwise RGB diff	**0.7311**	**0.8617**	**0.8753**	**0.8461**

Table G. Comparisons of pairwise and non-pairwise flow and RGB differences on the validation set of Kinetics-GEBD, measured by F1 score at different Rel.Dis. thresholds.

Model	0.05	0.25	0.5	Average
CSN + FC	0.7933	0.8954	0.9074	0.8834
CSN + DDM-Net	**0.8128**	**0.9077**	**0.9218**	**0.8972**

Table H. Performance of DDM-Net with CSN backbone on the validation set of Kinetics-GEBD, measured by F1 score at different Rel.Dis. thresholds.

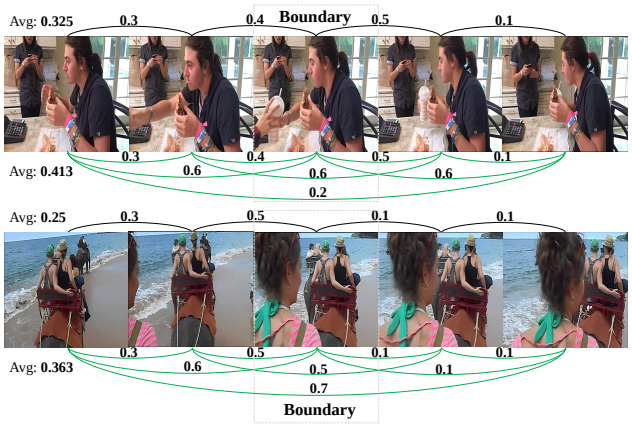
Time analysis. Average time cost of DDM-Net (0.123s) is close to [12] (0.066s). We calculate pairwise differences of frames in a sliding window (T frames, the sliding stride is s) rather than the whole video of E frames. Hence, the run-time complexity of DDM in a video is $O((E/s) \times T \times T)$ instead of $O(E \times E)$. In practice, T could be much smaller than E (*e.g.*, E = 300, T = 11, s = 3). Furthermore, computations can be re-used for subsequent frames since sliding windows are overlapped.

Complete Results of Precision, Recall and F1 score. Complete results of precision, recall and F1 score are presented in Table I and Table J. It is noteworthy that several methods (SceneDetect [1], PA [12]) achieve high precision yet very low recall, while several methods (TCN [7]) obtain high recall yet low precision. The first type of methods (high precision yet very low recall) focus on salient boundaries (*e.g.*, shot changes) and miss other event boundaries, while the second type of methods (high recall yet low precision) make as many predictions as possible and recall many false positives. As a result, both of them do not achieve superior F1 score.

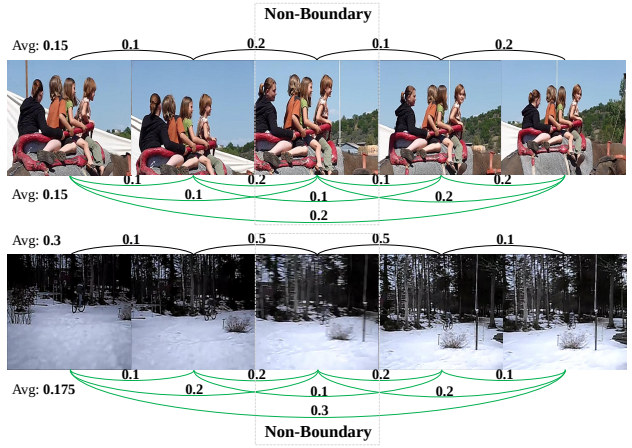
As for competitive methods [4, 6, 13] of LOVEU challenge, they do not present results on the standard validation set of Kinetics-GEBD for fair comparisons. Therefore, we compare DDM-Net with them on the testing set of Kinetics-GEBD in Table 3 of main paper (evaluated on the server of the challenge).

C. Visualization

More Comparisons of Sparse and Dense Motion Representation. We add more comparisons of sparse and



(a) Sparse and dense motion representations of boundary examples.



(b) Sparse and dense motion representations of non-boundary examples.

Figure A. More comparisons of sparse motion representation (black lines, optical flow) and dense motion representation (green lines, some are omitted for clarity, dense feature differences). Numbers on lines indicate the magnitude of motion between two frames. Dense motion representation provides more holistic temporal cues to better distinguish boundaries and non-boundaries.

dense motion representation, as displayed in Figure A. Figure A(a) illustrates two kinds of boundaries. The example in the first row is an 'event A→event B→event A' boundary (content of the boundary frame is different from other frames), while the second one is an 'event A→event B' boundary (the boundary frame is one frame of event A or event B). In both cases, the average magnitude of dense motion representation is larger than sparse motion representation, enabling the model to detect boundaries more easily. Figure A(b) also displays two kinds of non-boundaries. The first one is a non-boundary without large temporal changes, while the example in the second row is a non-boundary with temporal noise (camera blur). As our proposed DDM is calculated upon multi-level features instead of raw frames, it is more robust to noise than optical flow. Hence, the average magnitude of dense motion representation is smaller than sparse motion representation. Comparing the average magnitude in Figure A(a) and Figure A(b), we observe that holistic temporal clues of dense motion representation enable the model to better distinguish boundaries and non-boundaries.

Visualization of Progressive Attention Module. To further explore the effects of Progressive Attention Module, we present attention weight maps of intra-modal attention module and cross-modal attention module. In both Figure B and Figure C, differences between *columns* are significant, indicating that features of several moments or queries are enhanced. Figure B displays $T \times \omega$ cross-attention weight maps of intra-modal attention module. We observe that weight maps of non-boundaries are similar, as shown in the right subfigure. Since there are no obvious temporal changes in non-boundaries, attention weights of queries approximately follow Gaussian distribution (the weight decreases from the center frame to both sides). In contrast, weight maps of boundaries are diverse (left 2 subfigures), where queries attend to moments where temporal changes happen. Weight map of the last cross-attention layer in cross-modal attention module is shown in Figure C, where features of several queries are enhanced under the cross-modality guidance.

More Qualitative Results. We add more qualitative results to demonstrate the effectiveness of our proposed DDM-Net, as illustrated in Figure D. According to those examples, we conclude that DDM-Net predicts fewer false positives (high precision) and hits more ground truths (high recall) than PC [12], thus obtains a superior F1 score.

D. More Implementation Details

Complete Loss Function. We only present one classification loss function in the main paper because of the limited space. In practice, the loss function is the sum of 3 binary classification losses:

$$
\begin{aligned}
\mathcal{L}_{bc} &= \frac{1}{N}\sum_{\eta=1}^{N}(\mathcal{L}_{fu,\eta} + \mathcal{L}_{A,\eta} + \mathcal{L}_{D,\eta}), \\
\mathcal{L}_{fu,\eta} &= -(\hat{p}_\eta \log p_{fu,\eta} + (1-\hat{p}_\eta)\log(1-p_{fu,\eta})), \\
\mathcal{L}_{A,\eta} &= -(\hat{p}_\eta \log p_{A,\eta} + (1-\hat{p}_\eta)\log(1-p_{A,\eta})), \\
\mathcal{L}_{D,\eta} &= -(\hat{p}_\eta \log p_{D,\eta} + (1-\hat{p}_\eta)\log(1-p_{D,\eta})),
\end{aligned}
\tag{A}
$$

where $p_{fu,\eta}$, $p_{A,\eta}$ and $p_{D,\eta}$ are binary classification probabilities of fusion logit l, appearance logit l_A and difference logit l_D of the sample. N is the total number of training samples. $\hat{p}_\eta$ is 1 if the sample is marked as a boundary, and otherwise 0.

Detailed Formulas of Progressive Attention Module. Due to the limited space of the main paper, we define $\mathbf{q}$, $\mathbf{k}$ and $\mathbf{v}$ of intra-modal attention module and cross-modal attention module respectively. In this section, we present the

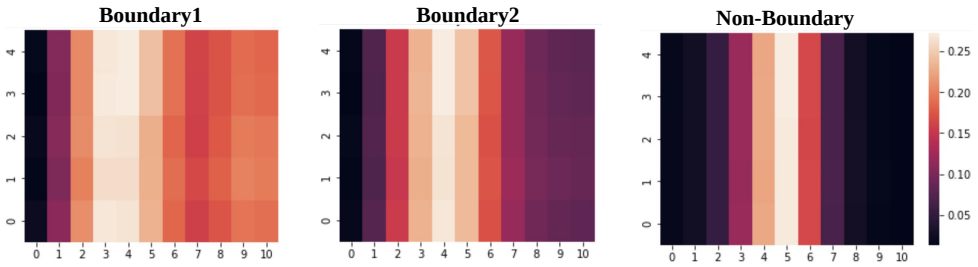


Figure B. Visualization of cross-attention weight maps in the intra-modal attention module, averaged among multiple heads of the last cross-attention layer. The y-axis is event queries and the x-axis represents timestamps of features. The color represents the magnitude of weight, as the weight becomes larger from black to white. Best viewed in color.

(a) Precision

Rel.Dis. threshold		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	avg
Unsuper.	SceneDetect [1]	0.731	0.792	0.819	0.837	0.847	0.856	0.862	0.867	0.870	0.872	0.835
	PA - Random [12]	0.737	0.884	0.933	0.956	0.968	0.975	0.979	0.981	0.984	0.986	0.938
	PA [12]	0.836	0.944	0.965	0.973	0.978	0.980	0.983	0.985	0.986	0.989	0.962
Super.	BMN [9]	0.128	0.141	0.148	0.152	0.156	0.159	0.162	0.164	0.165	0.167	0.154
	BMN-StartEnd [9]	0.396	0.479	0.509	0.525	0.534	0.540	0.544	0.547	0.549	0.551	0.517
	TCN-TAPOS [7]	0.518	0.622	0.665	0.690	0.706	0.718	0.727	0.733	0.738	0.743	0.686
	TCN [7]	0.461	0.519	0.538	0.547	0.553	0.557	0.559	0.561	0.563	0.564	0.542
	PC [12]	0.624	0.753	0.794	0.816	0.828	0.836	0.841	0.844	0.846	0.849	0.803
	DDM-Net	0.732	0.812	0.836	0.849	0.856	0.860	0.863	0.865	0.867	0.869	0.841

(b) Recall

Rel.Dis. threshold		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	avg
Unsuper.	SceneDetect [1]	0.170	0.185	0.192	0.197	0.200	0.202	0.204	0.206	0.207	0.207	0.197
	PA - Random [12]	0.218	0.289	0.326	0.350	0.364	0.374	0.381	0.386	0.389	0.393	0.347
	PA [12]	0.259	0.329	0.355	0.368	0.377	0.382	0.386	0.390	0.392	0.395	0.363
Super.	BMN [9]	0.338	0.369	0.385	0.397	0.407	0.414	0.420	0.426	0.430	0.434	0.402
	BMN-StartEnd [9]	0.648	0.766	0.817	0.846	0.864	0.876	0.885	0.892	0.897	0.900	0.839
	TCN-TAPOS [7]	0.420	0.508	0.550	0.576	0.594	0.609	0.619	0.627	0.633	0.639	0.577
	TCN [7]	0.811	0.894	0.923	0.938	0.947	0.952	0.956	0.959	0.961	0.963	0.930
	PC [12]	0.626	0.764	0.814	0.843	0.859	0.871	0.879	0.885	0.889	0.892	0.832
	DDM-Net	0.800	0.875	0.899	0.912	0.920	0.926	0.930	0.933	0.935	0.937	0.907

(c) F1

Rel.Dis. threshold		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	avg
Unsuper.	SceneDetect [1]	0.275	0.300	0.312	0.319	0.324	0.327	0.330	0.332	0.334	0.335	0.318
	PA - Random [12]	0.336	0.435	0.484	0.512	0.529	0.541	0.548	0.554	0.558	0.561	0.506
	PA [12]	0.396	0.488	0.520	0.534	0.544	0.550	0.555	0.558	0.561	0.564	0.527
Super.	BMN [9]	0.186	0.204	0.213	0.220	0.226	0.230	0.233	0.237	0.239	0.241	0.223
	BMN-StartEnd [9]	0.491	0.589	0.627	0.648	0.660	0.668	0.674	0.678	0.681	0.683	0.640
	TCN-TAPOS [7]	0.464	0.560	0.602	0.628	0.645	0.659	0.669	0.676	0.682	0.687	0.627
	TCN [7]	0.588	0.657	0.679	0.691	0.698	0.703	0.706	0.708	0.710	0.712	0.685
	PC [12]	0.625	0.758	0.804	0.829	0.844	0.853	0.859	0.864	0.867	0.870	0.817
	DDM-Net	**0.764**	**0.843**	**0.866**	**0.880**	**0.887**	**0.892**	**0.895**	**0.898**	**0.900**	**0.902**	**0.873**

Table I. Precision, Recall and F1 score of state-of-the-art GEBD methods on Kinetics-GEBD.

(a) Precision

Rel.Dis. threshold		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	avg
Unsuper.	SceneDetect	0.391	0.506	0.532	0.576	0.596	0.608	0.621	0.628	0.641	0.647	0.575
	PA - Random [12]	0.206	0.304	0.356	0.404	0.432	0.452	0.466	0.481	0.491	0.500	0.409
	PA [12]	0.470	0.599	0.662	0.708	0.740	0.755	0.771	0.784	0.795	0.801	0.708
Super.	ISBA [2]	0.119	0.185	0.230	0.268	0.301	0.329	0.356	0.379	0.392	0.405	0.296
	TCN [7]	0.140	0.187	0.200	0.204	0.207	0.208	0.210	0.211	0.211	0.211	0.199
	CTM [5]	0.154	0.197	0.212	0.221	0.228	0.233	0.237	0.242	0.244	0.245	0.221
	TransParser [11]	0.230	0.302	0.345	0.377	0.398	0.410	0.420	0.427	0.432	0.437	0.378
	PC [12]	0.650	0.741	0.782	0.805	0.821	0.829	0.836	0.842	0.846	0.851	0.800
	DDM-Net	0.591	0.667	0.700	0.720	0.732	0.737	0.741	0.744	0.748	0.751	0.713

(b) Recall

Rel.Dis. threshold		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	avg
Unsuper.	SceneDetect	0.018	0.023	0.025	0.027	0.028	0.028	0.029	0.029	0.030	0.030	0.027
	PA - Random [12]	0.128	0.189	0.221	0.252	0.269	0.281	0.290	0.299	0.305	0.311	0.255
	PA [12]	0.292	0.372	0.412	0.440	0.460	0.470	0.480	0.488	0.494	0.498	0.441
Super.	ISBA [2]	0.095	0.158	0.225	0.263	0.296	0.323	0.340	0.360	0.373	0.386	0.282
	TCN [7]	0.757	0.940	0.974	0.985	0.989	0.990	0.994	0.994	0.994	0.994	0.961
	CTM [5]	0.596	0.752	0.811	0.843	0.860	0.875	0.886	0.894	0.898	0.901	0.831
	TransParser [11]	0.386	0.516	0.590	0.642	0.673	0.689	0.705	0.714	0.721	0.726	0.636
	PC [12]	0.436	0.497	0.525	0.541	0.551	0.556	0.561	0.565	0.568	0.572	0.537
	DDM-Net	0.617	0.695	0.730	0.751	0.764	0.769	0.774	0.777	0.780	0.783	0.744

(c) F1

Rel.Dis. threshold		0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	avg
Unsuper.	SceneDetect	0.035	0.045	0.047	0.051	0.053	0.054	0.055	0.056	0.057	0.058	0.051
	PA - Random [12]	0.158	0.233	0.273	0.310	0.331	0.347	0.357	0.369	0.376	0.384	0.314
	PA [12]	0.360	0.459	0.507	0.543	0.567	0.579	0.592	0.601	0.609	0.615	0.543
Super.	ISBA [2]	0.106	0.170	0.227	0.265	0.298	0.326	0.348	0.369	0.382	0.396	0.302
	TCN [7]	0.237	0.312	0.331	0.339	0.342	0.344	0.347	0.348	0.348	0.348	0.330
	CTM [5]	0.244	0.312	0.336	0.351	0.361	0.369	0.374	0.381	0.383	0.385	0.350
	TransParser [11]	0.289	0.381	0.435	0.475	0.500	0.514	0.527	0.534	0.540	0.545	0.474
	PC [12]	0.522	0.595	0.628	0.646	0.659	0.665	0.671	0.676	0.679	0.683	0.642
	DDM-Net	**0.604**	**0.681**	**0.715**	**0.735**	**0.747**	**0.753**	**0.757**	**0.760**	**0.763**	**0.767**	**0.728**

Table J. Precision, Recall and F1 score of state-of-the-art GEBD methods on TAPOS.

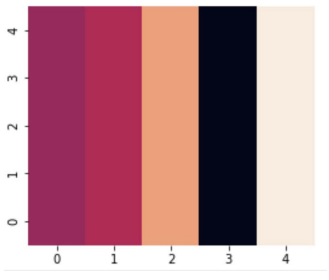


Figure C. Visualization of cross-attention weight maps in the cross-modal attention module, averaged among multiple heads of the last cross-attention layer. The y-axis and the x-axis represent event queries. Best viewed in color.

complete inference process of attention modules with detailed formulas. First, we review Attention and Multi-Head Attention mechanism [15],

$$\begin{aligned} \text{Attn}(\mathbf{q},\mathbf{k},\mathbf{v}) &= \text{softmax}(\frac{\mathbf{q}\mathbf{k}^\intercal}{\sqrt{d_k}})\mathbf{v}, \\ \text{MHA}(\mathbf{q},\mathbf{k},\mathbf{v}) &= \text{Concat}(\text{head}_1,\cdots,\text{head}_\mathbf{H})\mathbf{W}^O, \\ \text{head}_i &= \text{Attn}(\mathbf{q}\mathbf{W}_i^\mathbf{q},\mathbf{k}\mathbf{W}_i^\mathbf{k},\mathbf{v}\mathbf{W}_i^\mathbf{v}), \end{aligned} \tag{B}$$

where d_k is the dimension of features and $\mathbf{W}$ is the learnable projection matrix to transform the feature. Then, the calculation of each layer in intra-modal attention module

and cross-modal attention module can be formulated as:

$$\begin{aligned} \mathbf{q}' &= \mathrm{LN}(\mathbf{q} + \mathrm{MHA}(\mathbf{q}, \mathbf{q}, \mathbf{q})), \\ \mathbf{q}'' &= \mathrm{LN}(\mathbf{q}' + \mathrm{MHA}(\mathbf{q}', \mathbf{k}, \mathbf{v})), \\ \mathrm{Output} &= \mathrm{LN}(\mathbf{q}'' + \mathrm{FFN}(\mathbf{q}'')), \end{aligned} \tag{C}$$

where LN and FFN denote layer normalization and feed forward network.

References

[1] Brandon Castellano. PySceneDetect: an intelligent scene cut detection and video splitting tool. https://github.com/Breakthrough/PySceneDetect, 2014. 2, 4

[2] Li Ding and Chenliang Xu. Weakly-supervised action segmentation with iterative soft boundary assignment. In *CVPR*, pages 6508–6516. Computer Vision Foundation / IEEE Computer Society, 2018. 5

[3] Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *CVPR*, pages 12046–12055. Computer Vision Foundation / IEEE, 2019. 2

[4] Dexiang Hong, Congcong Li, Longyin Wen, Xinyao Wang, and Libo Zhang. Generic event boundary detection challenge at cvpr 2021 technical report: Cascaded temporal attention network (castanet), 2021. 2

[5] De-An Huang, Li Fei-Fei, and Juan Carlos Niebles. Connectionist temporal modeling for weakly supervised action labeling. In *ECCV (4)*, volume 9908 of *Lecture Notes in Computer Science*, pages 137–153. Springer, 2016. 5

[6] Hyolim Kang, Jinwoo Kim, Kyungmin Kim, Taehyun Kim, and Seon Joo Kim. Winning the cvpr'2021 kinetics-gebd challenge: Contrastive learning approach, 2021. 2

[7] Colin Lea, Austin Reiter, René Vidal, and Gregory D. Hager. Segmental spatiotemporal cnns for fine-grained action segmentation. In *ECCV (3)*, volume 9907 of *Lecture Notes in Computer Science*, pages 36–52. Springer, 2016. 2, 4, 5

[8] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 936–944. IEEE Computer Society, 2017. 1

[9] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. BMN: boundary-matching network for temporal action proposal generation. In *ICCV*, pages 3888–3897. IEEE, 2019. 4

[10] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *ECCV (1)*, volume 9905 of *Lecture Notes in Computer Science*, pages 21–37. Springer, 2016. 1

[11] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Intra- and inter-action understanding via temporal action parsing. In *CVPR*, pages 727–736. Computer Vision Foundation / IEEE, 2020. 5

[12] Mike Zheng Shou, Stan Weixian Lei, Weiyao Wang, Deepti Ghadiyaram, and Matt Feiszli. Generic event boundary detection: A benchmark for event segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8075–8084, October 2021. 1, 2, 3, 4, 5

[13] Quan Sun, Shimin Chen, Chen Chen, Xunqiang Tao, and Yandong Guo. Generic event boundary detection: Submission to loveu challenge 2021. https://github.com/VisualAnalysisOfHumans/LOVEU_TRACK1_TOP3_SUBMISSION, 2021. 2

[14] Du Tran, Heng Wang, Matt Feiszli, and Lorenzo Torresani. Video classification with channel-separated convolutional networks. In *ICCV*, pages 5551–5560. IEEE, 2019. 2

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 5

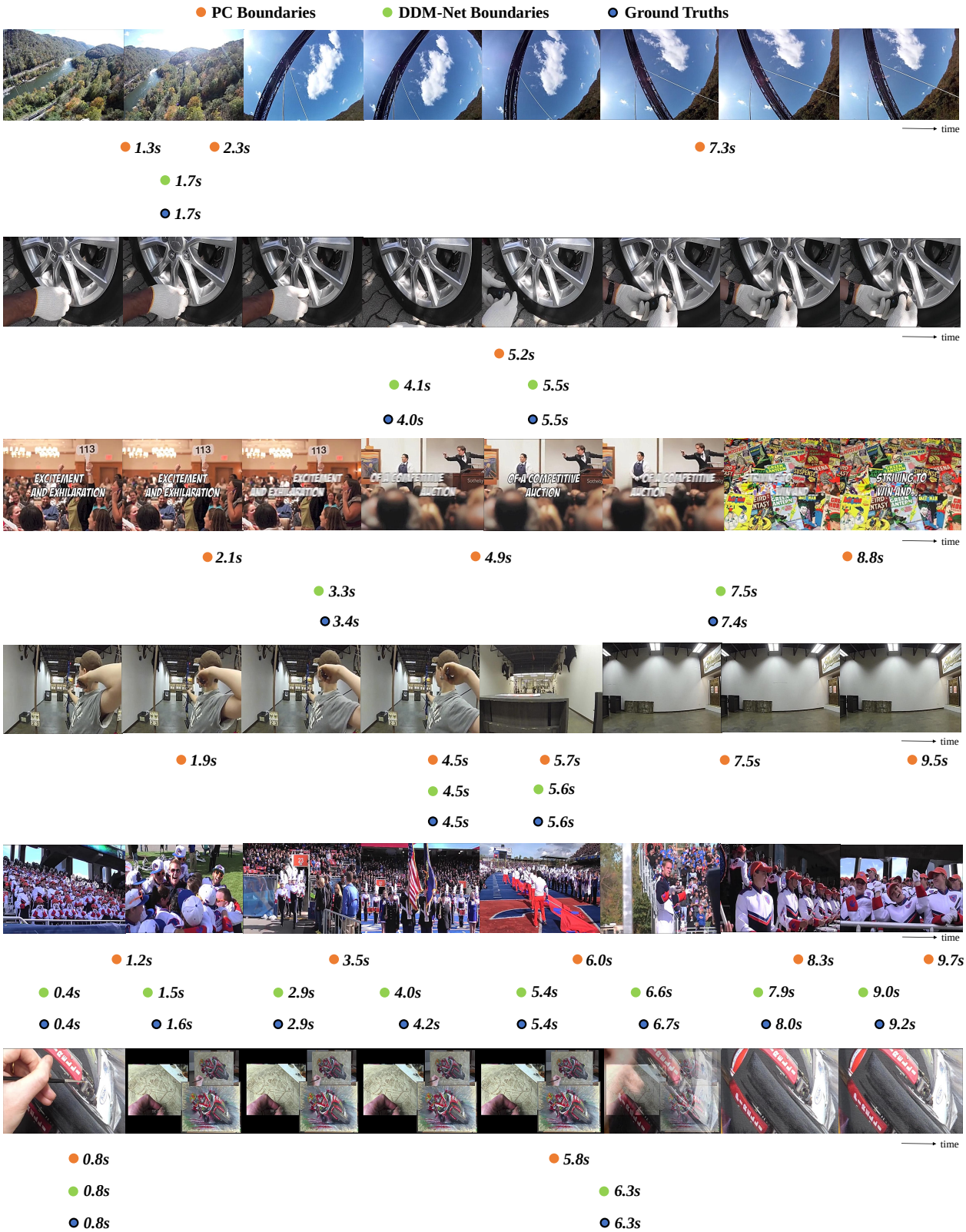


Figure D. More qualitative results and comparisons of PC, DDM-Net and ground truths on Kinetics-GEBD dataset.