Segment Any Event Streams via Weighted Adaptation of Pivotal Tokens
Supplementary Material

This supplementary material contains the following contents. We illustrate the motivation for developing this event-centric object segmentation network in Sec. S1. In Sec. S2, we also theoretically prove and experimentally evaluate the effectiveness of the utilized approximation strategy. We also compare our fine-tuning MLPs strategy with low-rank adaptation in Sec. S3, and explore the impact of weight initialization in Sec. S4. We also experimentally compare the proposed weighted distillation with other distillation methods, e.g., affinity graph-based regularization in Sec. S5. We experimentally evaluate the network performance with different event representations in Sec. S6 and different hyperparameters in Sec. S7. In Sec. S8 and S9, we make adaptation of huge ViT-backbone and intergrate it with large language models. Moreover, we have supplemented code and video for a better visual demonstration (https://github.com/happychenpipi/EventSAM).

S1. Motivation & Implementation Details for Event-Centric Universal Object Segmentation Networks

The manuscript, as delineated in the introductory section, focuses on integrating the distinctive attributes of event data—such as high dynamic range and temporal resolution—with the robust object recognition capabilities of current large pre-trained models. Consequently, the selection of image-event pairs in this work is specifically those free from degradation, for instance, those captured under slow motion and optimal illumination conditions. This approach is applied to calibrate the event embeddings to the image embeddings during both the training and evaluation processes. While the authors acknowledge the inevitable presence of errors in the outputs of image-fed Semantic Attention Models (SAMs) employed as labels during evaluation and training, it should also be noted that recent experiments have demonstrated the robust generalization abilities of SAMs. Their predictions have been found to align well with the requirements of visual recognition [1, 3, 5, 7]. Therefore, the evaluation of our methods primarily concentrates on the consistency of event-centric SAMs with the original SAMs, a method deemed both reasonable and effective.

Moreover, to evaluate the effectiveness of the proposed method for achieving object segmentation in extreme environments, we manually selected a subset of degraded scenes that contain low-light, overexposure, and motion blur. We compare the proposed method with the image-fed SAM, reconstructed frame-based event.

The experimental results are shown in Fig. S1, and we could figure out that the event-centric SAMs could indeed generalize to those extreme scenarios, validating the necessity of building such a pipeline. Moreover, our method could also outperform the frame reconstruction-based method in some cases, e.g., the 2^{nd} and 4^{th} rows, which further indicates the necessity of such an event-centric SAM.

S1.1. Implementation Details

Representation. To make events compatible with the RGB domain, following the setting in [8], we aggregate the events into a three-dimensional event volume $V \in \mathbb{R}^{H\times W\times B}$. For events in a time interval $[T, T+\delta_T]$, we uniformly divide events into $B+1$ time bins, then events in $[T+\frac{(i-1)\times\delta_T}{B+1}, T+\frac{(i+1)\times\delta_T}{B+1}]$ are integrated into i^{th} volume channel, where $i \in \{1, \cdots, B\}$. Note that the timestamp of corresponding RGB image is T. In our experiment, we set $B = 3$ and $\delta_T = 40ms$. Please refer to Sec. S6 for more analysis of representation.

Metric Calculation. For the calculation of area-weighted IOU (aIOU) in this manuscript, we first calculate the IOU of each segment for the event-centric SAM. Then we apply weighted summation of the IOU of each segment by it's area ratio in the image.

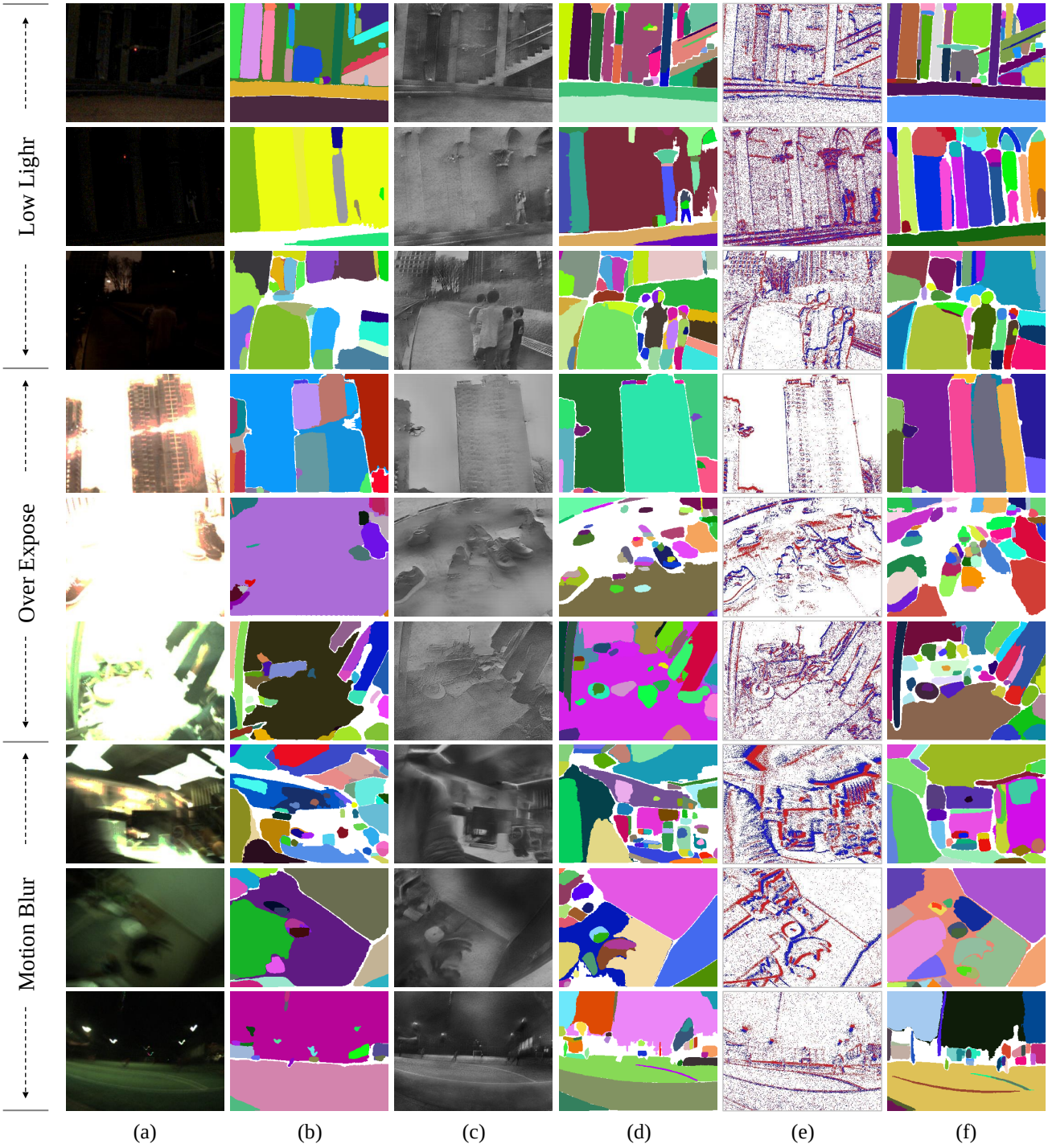

Figure S1. Visual comparison of the segmentation results from different degraded scenes, where each column indicate as the following: (a) the degraded image, (b) SAM output w/ image, (c) the reconstructed image by event data (E2VID [4]), (d) SAM output w/ reconstructed image, (e) corresponding event data, (f) SAM adapted w/ Ours.

S2. Approximation of Regularization Weights $\widehat{\mathbf{H}}^{(s)}$

In the manuscript, we formulate the transition process as follows:

$$\mathbf{H}^{(s)} = \prod_{i=s}^{n}[\alpha_i \mathbf{P}^{(i)} + (1-\alpha_i)\mathbf{I}], \tag{S1}$$

where $\mathbf{H}^{(s)} \in \mathbb{R}^{k_s \times k_{n+1}}$ symbolizes a comprehensive information transition matrix from a specific s^{th} layer to the terminal n^{th} layer. Moreover, the scalar α_i signifies the scaling influence of the normalization layers and MLPs. Its results are shown as follows:

$$\begin{aligned}\mathbf{H}^{(s)} =&(\alpha_s\alpha_{s+1}\cdots\alpha_n)\mathbf{P}^{(s)} \times \mathbf{P}^{(s+1)} \cdots \times \mathbf{P}^{(n)} + ((1-\alpha_s)\alpha_{s+1}\cdots\alpha_n)\mathbf{I} \times \mathbf{P}^{(s+1)} \times \cdots \times \mathbf{P}^{(n)} + \\ &(\alpha_s\alpha_{s+1}\cdots(1-\alpha_n))\mathbf{P}^{(s)} \times \mathbf{P}^{(s+1)} \times \cdots \times \mathbf{I} + ((1-\alpha_s)(1-\alpha_{s+1})\cdots\alpha_n)\mathbf{I} \times \mathbf{I} \times \cdots \times \mathbf{P}^{(n)} + \\ &\cdots + ((1-\alpha_s)(1-\alpha_{s+1})\cdots(1-\alpha_n))\mathbf{I} \times \mathbf{I} \times \cdots \times \mathbf{I}.\end{aligned} \tag{S2}$$

To thoroughly investigate such information flow, we take it as a transition process in markov chain. Here we define each $\mathbf{P}^{(i)}$ as a transition matrix $\mathbf{Q}^{(i)}$ from state i to $i+1$. If such the transition matrix has the following characteristics, the Markov chain would converge to a distribution.

$$\lim_{i \to +\infty} \mathbf{Q}^{(s)} * \mathbf{Q}^{(s+1)} * \cdots \mathbf{Q}^{(i)} \to \mathbf{M}, \tag{S3}$$

where $\mathbf{M}$ indicates the limitation of a series of matrix products. We assume that the attention matrix $\mathbf{P}^{(i)}$ has such convergence properties as Eq. S8 (as experimentally evaluated in Sec. S2.1). Thus, we have

$$\|\mathbf{P}^{(s)} * \mathbf{P}^{(s+1)} * \cdots \mathbf{P}^{(i)} - \mathbf{M}\|_2 < \delta, \quad \|\mathbf{P}^{(s)} * \mathbf{P}^{(s+1)} * \cdots \mathbf{P}^{(i+1)} - \mathbf{M}\|_2 < \delta \tag{S4}$$

where δ is a small scalar. Through triangle inequality, we have

$$\|\mathbf{P}^{(s)} \times \mathbf{P}^{(s+1)} \times \cdots \mathbf{P}^{(i)} - \mathbf{P}^{(s)} \times \mathbf{P}^{(s+1)} \times \cdots \mathbf{P}^{(i+1)}\|_2 \leq 2\delta, \tag{S5}$$

Moreover, with sub-multiplicative property of matrix norms, we have

$$\begin{aligned}\|\mathbf{P}^{(s)} \times \mathbf{P}^{(s+1)} \times \cdots \times \mathbf{P}^{(i)} \times \mathbf{P}^{(i+2)} \times \cdots \times \mathbf{P}^{(n)} - \mathbf{P}^{(s)} \times \mathbf{P}^{(s+1)} \times \cdots \mathbf{P}^{(i+1)} \times \mathbf{P}^{(i+2)} \times \cdots \times \mathbf{P}^{(n)}\|_2 \leq \\ \|\mathbf{P}^{(s)} \times \mathbf{P}^{(s+1)} \times \cdots \mathbf{P}^{(i)} - \mathbf{P}^{(s)} \times \mathbf{P}^{(s+1)} \times \cdots \mathbf{P}^{(i+1)}\|_2 \cdot \|\mathbf{P}^{(i+2)} \times \cdots \times \mathbf{P}^{(n)}\|_2 \leq 2k\delta,\end{aligned} \tag{S6}$$

where the $\|\mathbf{P}^{(i+2)} \times \cdots \times \mathbf{P}^{(n)}\|_2 \leq k$, where k is not a large number due to they act as transition matrices. Thus, randomly dropping several intermediate matrices would not greatly influence the transition process. Then we could approximate each term except the last term with $\prod_{i=s}^{n} \mathbf{P}^{(i)}$. It results in

$$\begin{aligned}\mathbf{H}^{(s)} &\approx (\alpha_s\alpha_{s+1}\cdots\alpha_n)\prod_{i=s}^{n}\mathbf{P}^{(i)} + ((1-\alpha_s)\alpha_{s+1}\cdots\alpha_n)\prod_{i=s}^{n}\mathbf{P}^{(i)} + \cdots + ((1-\alpha_s)(1-\alpha_{s+1})\cdots(1-\alpha_n))\mathbf{I} \\ &\approx \beta\prod_{i=s}^{n}\mathbf{P}^{(i)} + (1-\beta)\mathbf{I},\end{aligned} \tag{S7}$$

where we do not enforce $\beta = (\alpha_s\alpha_{s+1}\cdots\alpha_n) + ((1-\alpha_s)\alpha_{s+1}\cdots\alpha_n) + \cdots$ and $(1-\beta) = ((1-\alpha_s)(1-\alpha_{s+1})\cdots(1-\alpha_n))$. Since we expect through coordinate those parameters, it could compensate for some errors in the approximation process.

In this section, we conduct an experimental evaluation to assess the impact of varying values of the hyperparameter β. The results are presented in Table S1, where we systematically increase β from 0 to 1 in increments of 0.25, while keeping other conditions constant.

We observe that the network performance exhibits a gradual improvement, starting from 0.38 and reaching a saturation point at 0.41. As β increases, the performance of the network initially rises, but beyond a certain point, further increases in β lead to a decline in network performance. Similar results are also observed on the MVSEC dataset. Thus, we select β as 0.5.

Table S1. Ablation study results of the scaling factor β (see Eq. S7) based on RGBE-SEG and MVSEC datasets. Note that all methods are w/ the token mixing scheme.

β	RGBE-SEG				MVSEC			
	mP	mR	mIoU	aIoU	mP	mR	mIoU	aIoU
0.0	0.53	**0.74**	0.38	0.54	0.53	**0.71**	0.38	0.52
0.25	0.57	0.72	0.39	0.55	0.58	0.69	0.39	0.52
0.5	0.59	0.71	**0.41**	**0.55**	0.59	0.69	**0.40**	**0.52**
0.75	**0.61**	0.69	0.40	0.55	**0.61**	0.68	0.40	0.52
1.0	0.55	0.72	0.39	0.55	0.57	0.70	0.39	0.52

S2.1. Experimental Validation of the Assumption of Attention-transition

Our approximation is based on the assumption that the product of multiple layers' attention matrices will eventually converge to a matrix $\mathbf{M}$. In order to validate this assumption and examine the convergence properties, we calculate the L2 norm of the following quantities:

$$\mathcal{P}^i = \left\| \prod_{t=0}^{i} \mathbf{P}^{(t)} - \prod_{t=0}^{n} \mathbf{P}^{(t)} \right\|_2, \tag{S8}$$

where n indicates a total number of transition matrices in a ViT backbone.

The convergence of our approximation can be observed in Fig. S2, where the values of $\mathcal{P}^i$ decrease as i increases. This trend indicates that the product $\prod_{t=0}^{i} \mathbf{P}^{(t)}$ gradually approaches zero, supporting the notion that it converges to the desired matrix $\mathbf{M}$ as demonstrated in Eq. S1. The observed convergence provides validation for the correctness of our approximation. We also acknowledge that unrolling $\mathbf{H}^{(s)}$ leads to 2^i terms, each of which, when approximated, could contribute to a quantification error. To limit the cumulative error growth, the approach taken in this work is to consider only the subsequent three layers for each layer.

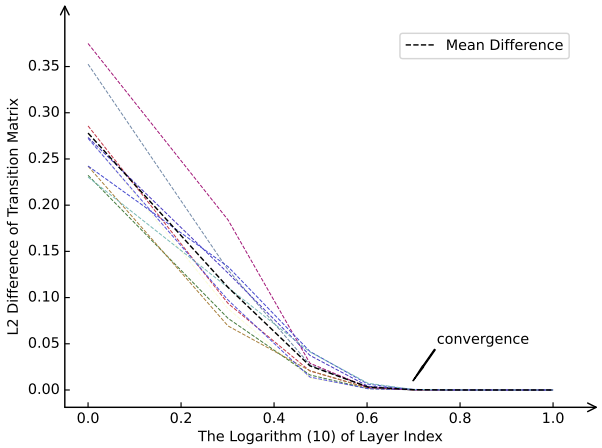

Figure S2. Illustration of the trends of $\mathcal{P}^i$ with variation of layer index i

S2.2. Necessity of such approximation

To validate the effectiveness of our approximation strategy, we also conduct extensive experiments do directly adopt Eq. S1 to calculate token significance. To achieve this, we manually set all alpha as the same. Due to that for some α the training losses do not deline, we only report the alpha greater than 0.9. The results indicate that without the properly setting the each α, the weighted KD loss even degrades the network performance.

Table S2. Ablation study results of the scaling factor α (see Eq. S1) based on RGBE-SEG and MVSEC datasets. For the relatively small α, e.g., 0.8 , the training loss indeed does not decline. Thus, we only report the experimental results with relatively large α. Note that all methods are w/ token mixing.

α	RGBE-SEG				MVSEC			
	mP	mR	mIoU	aIoU	mP	mR	mIoU	aIoU
0.99	0.39	0.80	0.31	0.50	0.41	0.80	0.31	0.52
0.95	0.38	0.81	0.30	0.50	0.41	0.80	0.31	0.51
0.90	0.38	0.79	0.29	0.49	0.40	0.80	0.31	0.51
Our baseline w/ token mixing	0.53	0.74	0.38	0.54	0.53	0.71	0.38	0.52

S3. Comparison with Low Rank Adaptation (LoRA) [2]

In the proposed method, we directly finetune the MLPs, instead of using some popular model adaptation tricks, e.g., LoRA. In this section, we conduct experiments to investigate our performance against LoRA. As shown in Table. S3, we try fine-tune the SAM with different methods. All methods are trained on the same dataset (RGBE-SEG training) and tested on RGBE-SEG and MVSEC, respectively. Note that the training configuration of those methods are exactly same, except the trainable parameters. The experimental results show that applying LoRA to the SAM adaptation cannot achieve considerable performance.

Table S3. Comparison of the segmentation performance between LoRA and our fine-tuning method based on RGBE-SEG and MVSEC datasets. And we have set up a series of ranks (r) of LoRA to fully explore its adaption effect. Note that all methods are w/o the token mixing and weighted regularization.

Fine-tuning Method	Trainable #Param	RGBE-SEG				MVSEC			
		mP	mR	mIoU	aIoU	mP	mR	mIoU	aIoU
w/o Fine-tuning	-	0.39	0.73	0.26	0.43	0.40	0.68	0.26	0.46
LoRA(Embed + Four MLPs, r=16)	1.1M	0.39	**0.76**	0.28	0.44	0.36	0.75	0.25	0.45
LoRA(Embed + Four MLPs, r=64)	2.6M	0.40	0.76	0.28	0.44	0.36	0.75	0.25	0.45
LoRA(Embed + Four MLPs, r=256)	8.5M	0.41	0.75	0.29	0.45	0.37	0.74	0.26	0.45
LoRA(Embed + All Blocks, r=16)	2.9M	0.40	0.75	0.27	0.43	0.35	**0.76**	0.24	0.44
LoRA(Embed + All Blocks, r=64)	10.0M	0.41	0.74	0.28	0.44	0.33	0.76	0.24	0.45
LoRA(Embed + All Blocks, r=256)	38.4M	0.40	0.74	0.28	0.44	0.34	0.76	0.24	0.45
Our baseline	29.0M	**0.52**	0.73	**0.37**	**0.53**	**0.53**	0.69	**0.37**	**0.52**

S4. Weight Initialization

In this section, we train the model by initializing pre-trained weights in different ratios, as detailed in Tab. S4, and we prioritized loading the weights of shallower layers. For the model initiated from scratch, we trained all weights. This observation aligns with the intuitive understanding that the weights of a large pre-trained model encapsulate extensive knowledge patterns from the pre-training dataset.

Table S4. Impact of weight initialization on RGBE-SEG dataset. Underline indicates our pre-trained weight setting in the paper. Note that all methods are w/ the token mixing and weighted regularization.

Pre-trained Weight Ratio	RGBE-SEG			
	mP	mR	mIoU	aIoU
0%	0.03	0.96	0.04	0.24
75%	0.05	0.92	0.07	0.28
100%	0.59	0.71	0.41	0.55

S5. Comparison with Affinity Graph KD

We also consider a comparison method for embedding KD, namely affinity graph-based KD [6]. In our evaluation, we focus on modifying the loss function to explore different knowledge distillation approaches while keeping other settings consistent. The experimental results, presented in Table S5, demonstrate that our proposed methods achieve superior improvements in terms of mIOU and maintain higher aIOU values. In contrast, the affinity graph-based KD approach exhibits a significant decrease in aIOU, indicating difficulties in achieving accurate segmentation over large areas. This observation is further supported by Fig. S3-(d), where the affinity graph-based method only successfully segments small regions and struggles to capture the global contour of objects.

Table S5. Comparison of the segmentation performance between affinity graph and our KD scheme based on RGBE-SEG and MVSEC datasets. Note that all methods are w/ the token mixing scheme.

KD Scheme	RGBE-SEG				MVSEC			
	mP	mR	mIoU	aIoU	mP	mR	mIoU	aIoU
Our baseline w/ token mixing	0.53	0.74	0.38	0.54	0.53	0.71	0.38	0.52
Affinity Graph	**0.67**	0.61	0.40	0.49	**0.61**	0.63	0.38	0.46
Incremental	+0.14	-0.13	+0.02	-0.05	+0.08	-0.07	0	-0.06
Ours	0.59	**0.71**	**0.41**	**0.55**	0.59	**0.69**	**0.40**	**0.52**
Incremental	+0.06	-0.03	+0.06	+0.01	+0.05	-0.02	0.02	0

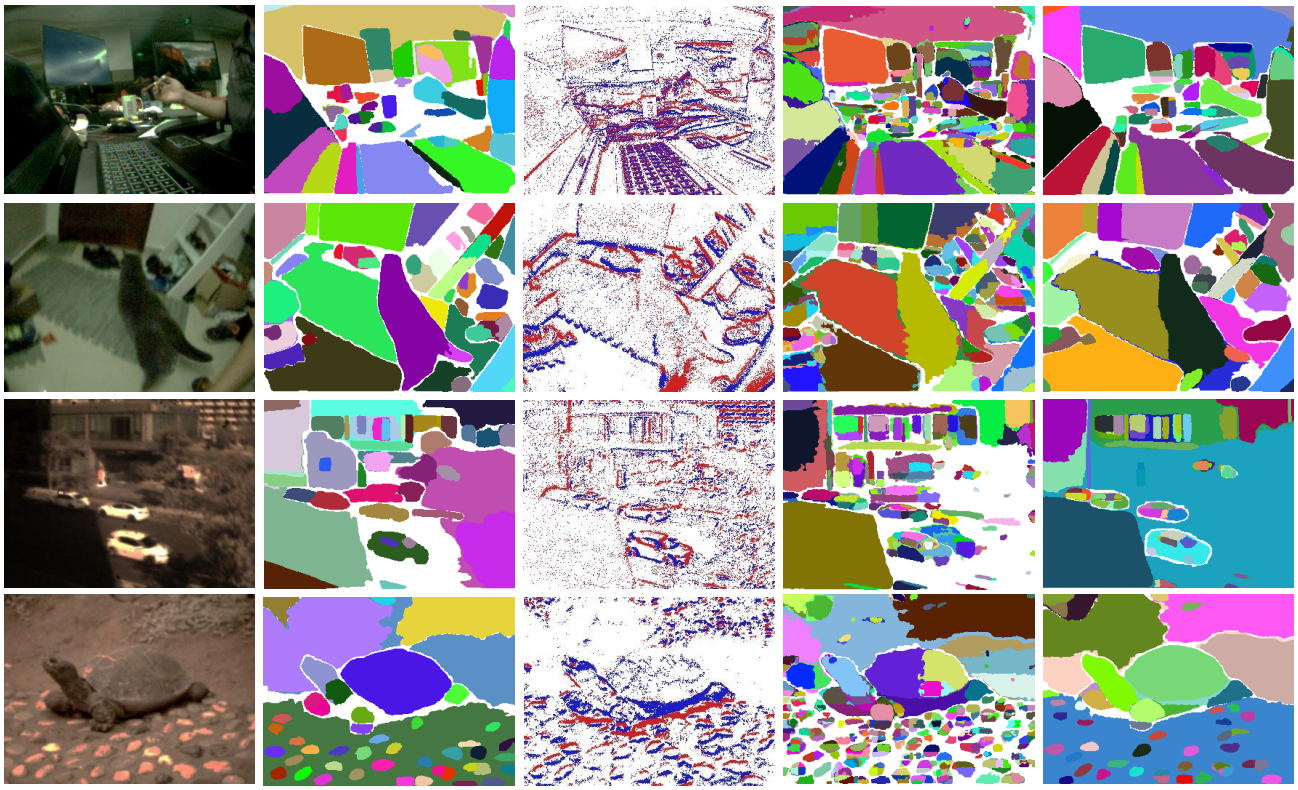

Figure S3. Visual comparison of the segmentation results from different KD manners, where each figure indicate as the following: (a) the reference image, (b) SAM output w/ image, (c) corresponding event data, (d) SAM adapted w/ affinity graph, (e) SAM adapted w/ Ours.

S6. Different Event Representation

We further conduct experiments to compare affect of different representations, including the time bin and time interval of event volume, as well as temporal modeling.

S6.1. Time Bin

We experimentally validate the network performance with different time bins B, where Table S6 lists the experimental results. The adopted $B = 3$ achieves the best performance. Note that for $B \leq 3$, event images are fed as RGB images. For $B > 3$, we modify the embedding layer to fit the input channel.

Table S6. Ablation study results of the time bin based on RGBE-SEG dataset. Underline indicates our time interval setting in the paper. Note that all methods are w/ the token mixing and weighted regularization.

time bin	RGBE-SEG			
	mP	mR	mIoU	aIoU
1	0.47	0.75	0.37	0.48
2	0.51	0.77	0.40	0.51
3	0.59	0.71	0.41	0.55
4	0.47	0.80	0.38	0.50
5	0.52	0.77	0.41	0.52

S6.2. Time Interval

We experimentally validate the network performance with different time interval δ_T, where Table S7 lists the experimental results. The adopted $\delta_T = 40ms$ achieves the best performance.

Table S7. Ablation study results of the time interval based on RGBE-SEG and MVSEC datasets. Underline indicates our time interval setting in the paper. Note that all methods are w/ the token mixing and weighted regularization.

time interval (ms)	RGBE-SEG				MVSEC			
	mP	mR	mIoU	aIoU	mP	mR	mIoU	aIoU
10	0.40	0.79	0.30	0.45	0.44	0.74	0.31	0.51
20	0.52	0.73	0.37	0.51	0.53	0.71	0.37	0.51
30	0.56	0.73	0.39	0.53	0.57	0.70	0.39	0.52
40	0.59	0.71	0.41	0.55	0.59	0.69	0.40	0.52
50	0.57	0.74	0.39	0.54	0.59	0.69	0.40	0.52
60	0.56	0.74	0.38	0.53	0.59	0.68	0.39	0.52

S6.3. Temporal Encoding

Further, we change the patch embedding layer in SAM with recurrent layer for better temporal modeling. However, experimental results (as shown in Table S8) show that after changing even the embeddings layer, the generalization ability of network gets serious degradation.

Table S8. Comparison of the segmentation metrics between SAM with recurrent and without recurrent modeling for feature embedding.

Embedding Layer	RGBE-SEG				MVSEC			
	mP	mR	mIoU	aIoU	mP	mR	mIoU	aIoU
w recurrent	0.39	0.74	0.27	0.45	0.42	0.78	0.32	0.49
w/o recurrent	0.59	0.71	0.41	0.55	0.59	0.69	0.40	0.52

S7. Hyperparameter of Training Objectives

In this section, we explore the performance with different hyperparameter γ_i. Intuitively, γ_i should increase with layer depth since the alignment of deeper features is more important. We experimented with variation of γ_i in Tab. S9, showing the rationality of our selections of γ.

Table S9. Comparison of different γ_i settings on RGBE-SEG. Underline indicates our hyperparameter setting in the paper. Note that all methods are w/ the token mixing and weighted regularization.

γ_{1-4}	RGBE-SEG			
	mP	mR	mIoU	aIoU
0.0, 0.3, 0.6, 1.0	0.54	0.70	0.38	0.53
0.2, 0.5, 0.8, 1.0	0.56	**0.72**	0.39	0.54
0.1, 0.4, 0.7, 1.0	**0.59**	0.71	**0.41**	**0.55**

S8. Adaptation of SAM with ViT-Huge

In this section, we further adapt a large SAM with ViT-H as backbone. The experimental results are shown as Fig. S4 and table S10. The ViT-H has stronger modeling ability than ViT-B. Thus, it easy for network ViT-H to approach the results from ViT-B,

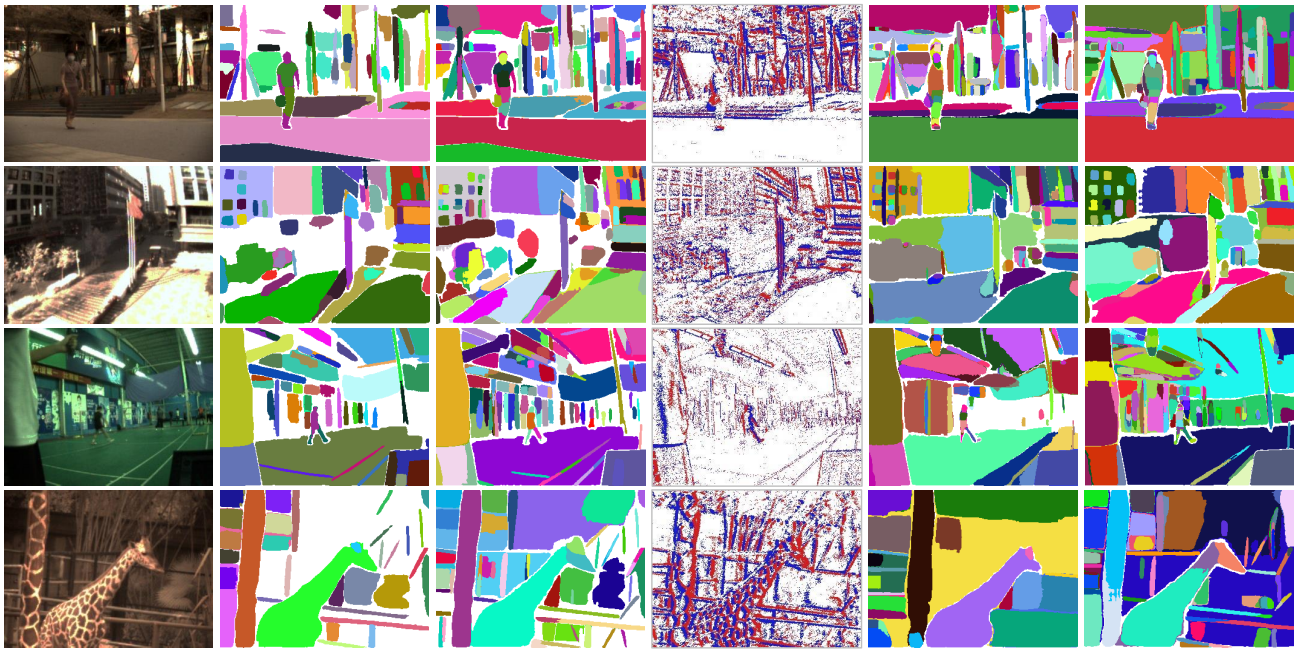

Figure S4. Visual comparison of the segmentation results from different SAM encoders, where each figure indicate as the following: (a) the reference image, (b) ViT-B SAM output w/ image, (c) ViT-H SAM output w/ image, (d) corresponding event data, (e) ViT-B SAM output w/ event data, (f) ViT-H SAM output w/ event data.

S9. Integration the Proposed Event-centric SAM with Large Language Models

To further validate the strong zero-shot object recognition ability of our event-adapt SAM. We integrate it with a vision-language object segmentation framework LISA [3]. The framework is shown as Fig. S5. Through this, we could further unlock the rich semantic inherent in SAM, for interactive universal object segmentation with Event data. Visual segmentation results are shown as Fig. S6. Please refer to supplementary video for more visualizations.

Table S10. Comparison of the segmentation metrics between SAM with ViT-Base and SAM with ViT-Huge with different reference masks. Underline indicates our setting in the paper.

Encoder	Reference Mask	RGBE-SEG				MVSEC			
		mP	mR	mIoU	aIoU	mP	mR	mIoU	aIoU
ViT-B	ViT-B	0.59	0.71	0.41	0.55	0.59	0.69	0.40	0.52
ViT-H	ViT-H	0.59	0.67	0.39	0.53	0.61	0.64	0.39	0.53
ViT-B	ViT-H	0.49	0.72	0.35	0.51	0.49	0.70	0.34	0.51
ViT-H	ViT-B	0.66	0.65	0.43	0.54	0.69	0.60	0.42	0.52

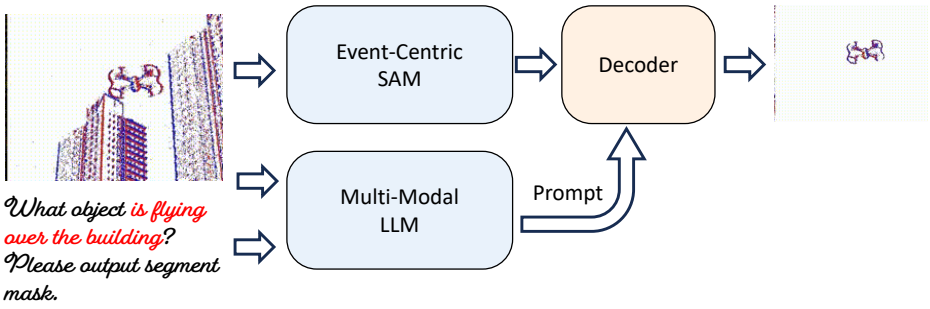

Figure S5. Illustration of integration of our event-centric SAM with LLM, where we utilize **LLaVA-13B-v1-1** as LLM backbone and **ViT-H** as the backbone of SAM.

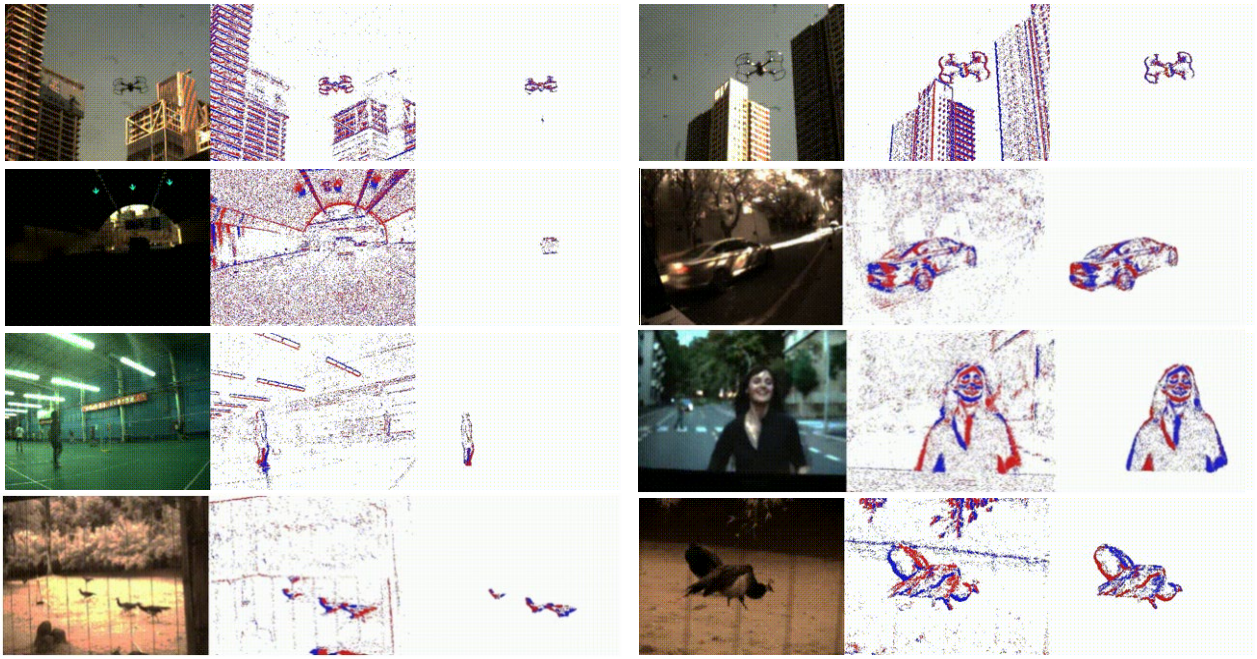

Figure S6. Segmentation results of language-prompt event segmentation, where (a) and (d) indicate the RGB images only for visualization not perceived by network, (b) and (e) represents input event, finally (c) and (f) for segmentation results.

References

[1] Ruining Deng, Can Cui, Quan Liu, Tianyuan Yao, Lucas W Remedios, Shunxing Bao, Bennett A Landman, Lee E Wheless, Lori A Coburn, Keith T Wilson, et al. Segment anything model (sam) for digital pathology: Assess zero-shot segmentation on whole slide imaging. *arXiv preprint arXiv:2304.04155*, 2023. 1

[2] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 5

[3] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. *arXiv preprint arXiv:2308.00692*, 2023. 1, 8

[4] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE TPAMI*, 43(6):1964–1980, 2019. 2

[5] Saikat Roy, Tassilo Wald, Gregor Koehler, Maximilian R Rokuss, Nico Disch, Julius Holzschuh, David Zimmerer, and Klaus H Maier-Hein. Sam. md: Zero-shot medical image segmentation capabilities of the segment anything model. *arXiv preprint arXiv:2304.05396*, 2023. 1

[6] Lin Wang, Yujeong Chae, and Kuk-Jin Yoon. Dual transfer learning for event-based end-task prediction via pluggable event to image translation. In *ICCV*, pages 2135–2145, 2021. 6

[7] Dingyuan Zhang, Dingkang Liang, Hongcheng Yang, Zhikang Zou, Xiaoqing Ye, Zhe Liu, and Xiang Bai. Sam3d: Zero-shot 3d object detection via segment anything model. *arXiv preprint arXiv:2306.02245*, 2023. 1

[8] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *CVPR*, pages 989–997, 2019. 1