Appendix

Supplementary Material

This appendix presents additional materials and results.

First, we describe the detailed training settings in Sec. A. Then, we give further descriptions of our proposed COMBO in Sec. B to enhance comprehension. Next, we provide more ablation studies for COMBO in Sec. C. Finally, a series of visual results are presented in Sec. D.

A. More Implementation Details

This section further explains the experimental details, which can be found in Tab. I. It should be noted that the batch size pertains to the number of videos entered, thereby implying $bs \times T$ frames per iteration, where bs denotes the batch size. Furthermore, pertaining to the AVSS task, given that the input video comprises varying numbers of frames, the number of frames within the batch size was dynamically altered without the need for padding zeros.

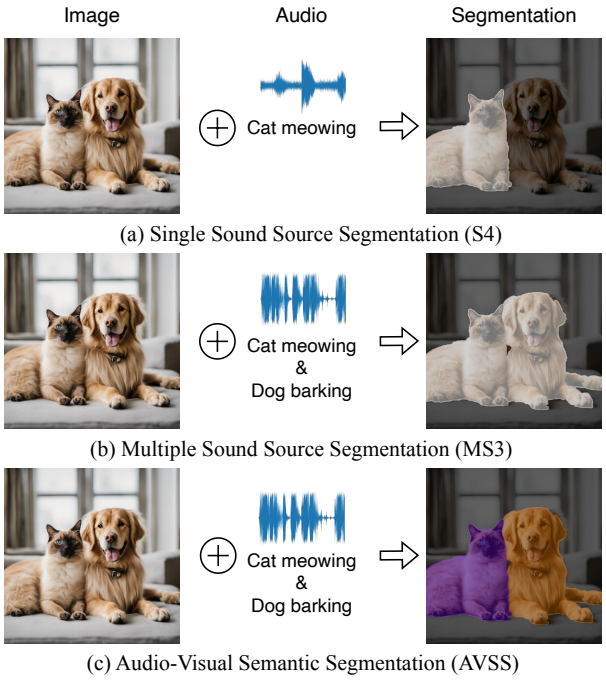
B. Further Descriptions

B.1. Task Description

We begin by providing an illustration description for the Audio-Visual Segmentation (AVS). As depicted in Fig. I, the purpose of AVS is to segment all sound objects pixel-by-pixel. There are two datasets included: (1) *AVSBench-object*. This dataset encompasses single source sound segmentation (S4) and multiple sound sources segmentation (MS3), as shown in Fig. I (a) and Fig. I (b), respectively. In other words, objects (such as a dog or cat) in an image can be categorized into class-agnostic masks based on their corresponding sounds. (2) *AVSBench-semantic*. In addition to the above, objects that emit sounds also carry class semantic information, a concept known as audio-visual semantic segmentation (AVSS), as shown in Fig. I (c). This represents a more challenging dataset due to its complexity.

B.2. Proposal Generator

As shown in Fig. II, we provide a more detailed explanation of the proposal generator proposed in COMBO. Initially, we obtain class-agnostic masks denoted as $c \in \mathbb{R}^{K\times H\times W}$ from the input frame, using a pre-existing foundation model [3], where K denotes the number of potential targets. Subsequently, a Maskige generator, which is part of the proposal generator, is introduced to convert the class-agnostic masks $c \in \mathbb{R}^{K\times H\times W}$ into Maskige, denoted as $m \in \mathbb{R}^{3\times H\times W}$ without the need for additional training. Particularly, as K is dynamic and fluctuates according to input frames, we amplify the quantity of class-agnostic masks to N using zero masks, thereby deriving a series of binary masks $c \in \mathbb{R}^{N\times H\times W}$, where N is a predetermined number and $N \geq K$. Next, considering that c denotes a series of binary masks that are challenging to incorporate into visual features, we utilize a random color encoding function $\mathcal{X}(\cdot) : \mathbb{R}^{N\times H\times W} \rightarrow \mathbb{R}^{3\times H\times W}$ to convert the binary masks $c \in \mathbb{R}^{N\times H\times W}$ into Maskige $m \in \mathbb{R}^{3\times H\times W}$. Formally, we define $\mathcal{X}(c) = cA$, where $A \in \mathbb{R}^{N\times 3}$. To facilitate the proposal generator offline, the value of A is manually set appropriately. Specifically, we set $N = 100$, a value



(a) Single Sound Source Segmentation (S4)

(b) Multiple Sound Source Segmentation (MS3)

(c) Audio-Visual Semantic Segmentation (AVSS)

Figure I. Illustration of the three sub-tasks in AVSBench-object and AVSBench-semantic datasets. Best viewed in color.

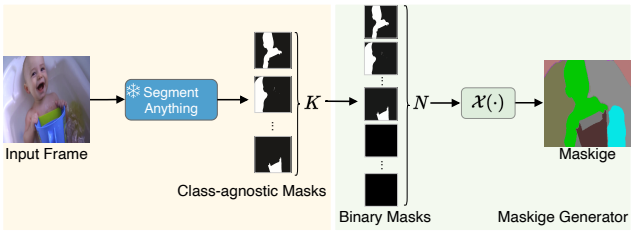


Figure II. Illustration of the Proposal Generator. The proposal generator consists of two parts: the yellow area on the left mainly contains a frozen foundation model for generating class-agnostic masks, and the green area on the right is used to convert the masks into Maskige, also called Maskige generator.

Settings	S4	MS3	AVSS
Resolution $H \times W$	224×224	224×224	224×224
Number of frames T	5	5	5 & 10
Data augmentation	horizontal flip & color aug	horizontal flip & color aug	horizontal flip & color aug
Audio dimension D	128	128	128
Embedding dimension d	256	256	256
Number of queries N_q	100	100	100
Number of transformer decoders L	3	3	3
Loss coefficient λ_{cls}	2.0	2.0	2.0
Loss coefficient λ_{mask}	5.0	5.0	5.0
Loss coefficient λ_{ada}	10.0	10.0	5.0
Batch size	8	8	8
Optimizer	AdamW	AdamW	AdamW
Learning rate	0.0001	0.0001	0.0001
Weight decay	0.05	0.05	0.05
Iterations	90k	20k	90k

Table I. Detailed settings. This table provides a detailed overview of the specific settings used for each sub-task.

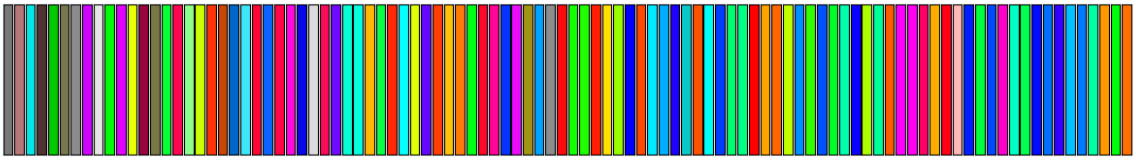


Figure III. Visualization of matrix $A \in \mathbb{R}^{100\times3}$. Each color bar has an RGB value of dimension 3, and there are 100 color bars.

considerably larger than K. To enhance the distinctiveness between various targets, as illustrated in Fig. III, we use the first 100 color mappings of the color mapping relationship in ADE20K dataset [6] as the parameters of matrix A. Further visualizations on Maskiges are available in Sec. D.

C. More Results

C.1. Effects of The Foundation Model

We continue our exploration by investigating the impact of the various foundation models in the proposal generator on performance. For comparison, we select the original Segment Anything Model (SAM) [2], the superior-performing Semantic-SAM [3], and the lighter MobileSAM [5] as the foundation models of the proposal generator to evaluate the performance alongside the backbone of PVT-v2 [4] on the S4 subset. As illustrated in Tab. II, the results depict a minimal performance discrepancy among the different foundational models. Nevertheless, it is evident that the performance of our model improves with the enhancement of the foundational model's ability. Accordingly, we choose the Semantic-SAM [3] as the foundation model of the proposal generator. In addition, we also provide a comparison of the visualizations of the Maskiges generated by different foundational models in Sec. D.

C.2. Effects of The Number of Queries

We present additional ablation studies concerning the number of queries, denoted as N_q, in our approach, as illustrated in Tab. III. In order to examine the influence of the query count on the model's performance, we conducted a series of experiments using varying quantities of queries within the transformer decoder, specifically 100, 200, and 300. Our findings suggest that 100 queries are sufficient, given the infrequency of maximum concurrent classes in an AVS task. Therefore, we established the default number of queries as 100 following [1].

S4	SAM [2]	Semantic-SAM [3]	MobileSAM [5]
$\mathcal{M}_\mathcal{J}$	84.4	**84.7**	84.1
$\mathcal{M}_\mathcal{F}$	91.8	**91.9**	91.6

Table II. Impact of the different foundation models on COMBO.

N_q	S4		AVSS	
	$\mathcal{M}_\mathcal{J}$	$\mathcal{M}_\mathcal{F}$	$\mathcal{M}_\mathcal{J}$	$\mathcal{M}_\mathcal{F}$
100	**81.7**	**90.1**	**33.3**	**37.3**
200	81.5	**90.1**	32.0	35.6
300	81.3	89.9	31.4	34.7

Table III. Impact of the number of queries on COMBO.

D. More Qualitative Results

In this section, we introduce additional qualitative results of our proposed COMBO, along with its intermediate visualizations, to illustrate the effectiveness of our module. The quality of the generation of the Maskige is crucial to the assistance of our model. Therefore, we first show some

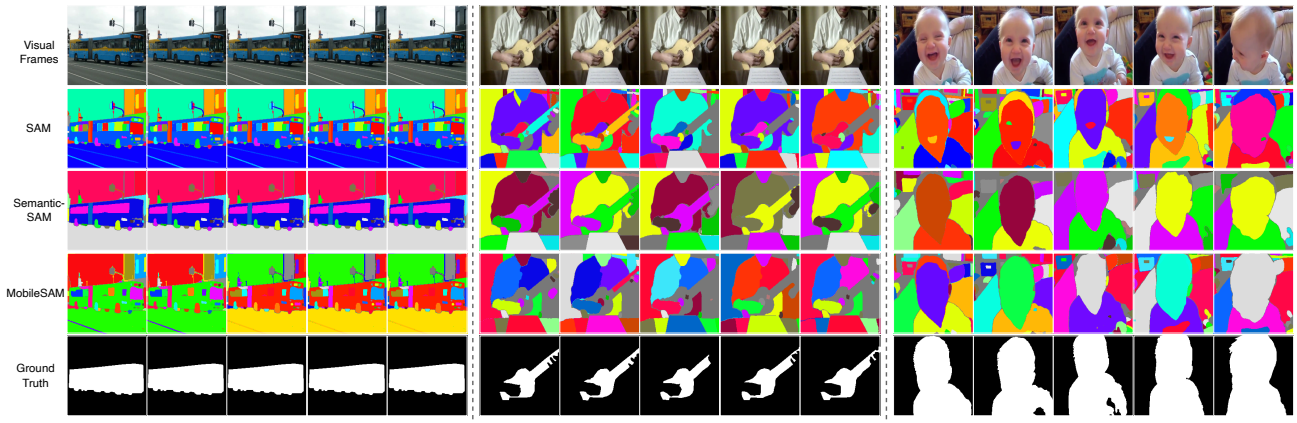


Figure IV. Visualization of Maskiges. The Maskiges are generated by proposal generator with various foundation models [2, 3, 5] and the color encoding function $\mathcal{X}(\cdot)$.

examples sampled from various sub-tasks with foundation models [2, 3, 5] in Fig. IV. It is evident that all foundational models exhibit exceptional proficiency in segmenting class-agnostic targets. However, given that Semantic-SAM [3] can produce a more complete target mask, we select it as the foundation model of our proposed proposal generator. Besides, we also provide additional heat maps of the predicted masks to illustrate the effectiveness of the adaptive inter-frame consistency loss, $\mathcal{L}_{ada}$. As depicted in Fig. V, when adjacent frames are similar, our loss module enables predicted masks to produce more accurate results. Conversely, when adjacent frames are dissimilar, our module can avoid mutual interference between adjacent frames due to the existence of adaptive. Finally, given that the audio-visual segmentation task is a video task with audio input, we present a visual comparison between our method and baseline in a video format, which can be reviewed on our project page.

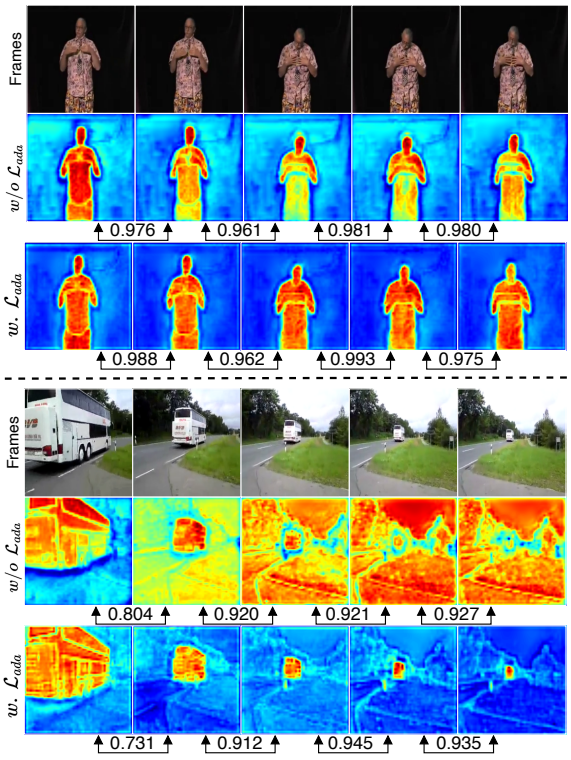


Figure V. Visualization of the heat map of the predicted masks without and with the consideration of $\mathcal{L}_{ada}$ based on the S4 subset.

References

[1] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1280–1289, 2022. 1

[2] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023. 1, 2

[3] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Jianwei Yang, Chunyuan Li, Lei Zhang, and Jianfeng Gao. Semantic-sam: Segment and recognize anything at any granularity. *arXiv preprint arXiv:2307.04767*, 2023. 0, 1, 2

[4] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022. 1

[5] Chaoning Zhang, Dongshen Han, Yu Qiao, Jung Uk Kim, Sung-Ho Bae, Seungkyu Lee, and Choong Seon Hong. Faster segment anything: Towards lightweight sam for mobile applications. *arXiv preprint arXiv:2306.14289*, 2023. 1, 2

[6] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2019. 1