

A. Object Extraction Prompt

Object Extraction Prompt

Input Expression

$\{Expression\}$

Prompt

What is the main object (or objects) referred to in the given expression or question?

Focus on the **primary subject or agent** involved in the described action or behavior.

Respond with a **single word** (e.g., “cat”, “person”, “dog”) that best describes the target object(s).

Figure 5. Prompt used to extract the target object from the referring expression.

To obtain informative visual attention maps from the frozen LVLM, we first augment the referring expression with a fixed object extraction prompt. Rather than directly prompting the model to localize the referred object, we ask it to identify the primary object category mentioned in the expression and generate a single response token (Figure 5). We then extract the attention of this generated token to all visual tokens across transformer layers and attention heads.

B. More Implementation Details

The training hyperparameters of the proposed attention adapter are summarized in Table 9. We optimize the network using AdamW with an initial learning rate of 10^{-4} that linearly decays to 10^{-5} over 20 epochs. A weight decay of 10^{-4} is applied for regularization, and training is performed with a batch size of eight video clips. The network is supervised using a combination of binary cross-entropy with logits and Soft Dice loss, which jointly encourage accurate pixel-wise classification and robust mask overlap.

Table 10 summarizes the architecture of the proposed attention adaptation network. The network follows a U-Net-style encoder-decoder design operating on concatenated frame- and video-level LVLM attention features. Given $C=392$ channels from each modality, the input contains $2C=784$ channels over a 24×42 spatial grid. The first encoder stage applies two 3×3 convolutions to produce 128 channels, followed by 2×2 max pooling. The second encoder stage increases the feature dimension to 256 channels at half the spatial resolution.

At the bottleneck, two additional 3×3 convolutions produce a 512-channel representation. To capture temporal dependencies across keyframes, the bottleneck feature of

Table 9. Training hyperparameters of the ATLAS adapter.

Hyperparameter	Value
Optimizer	AdamW
Learning rate	$10^{-4} \rightarrow 10^{-5}$ (linear decay)
Weight decay	10^{-4}
Epochs	20
Batch size	8 clips
Segmentation loss	BCE-with-Logits + Soft Dice

each frame is globally average pooled and passed through a two-layer, four-head temporal transformer. Its output is mapped to FiLM parameters γ and β , which modulate the 512-channel bottleneck features channel-wise. This conditions the spatial decoding of each frame on information from the full video sequence rather than processing frames independently.

The decoder then progressively restores the spatial resolution using transposed convolutions and U-Net skip connections. The first upsampling stage fuses the temporally modulated bottleneck with Encoder-2 features and reduces the representation to 256 channels. The second stage combines it with Encoder-1 features and restores the original $R_f \times W_f$ resolution with 128 channels. Finally, a 1×1 convolution produces a single-channel localization logit map, which provides object-centric spatial cues for the frozen SAM2 model.

C. Training Dynamics

Figure 6 shows stable adapter training, with the total, binary cross-entropy, and Soft Dice losses decreasing consistently. Qualitatively, the adapter progressively transforms diffuse LVLM attentions into object-centric localization maps by suppressing background responses and concentrating on the referred object, producing accurate cues for the frozen SAM2 model.

D. Additional Qualitative Results

Figure 7-15 provide additional qualitative examples, demonstrating that ATLAS consistently produces object-centric attention maps that guide accurate segmentation.

E. Failure cases

We add a quantitative failure analysis on ReasonVOS using Qwen2.5-VL-7B. Failures ($\mathcal{J} \& \mathcal{F} < 0.5$; 149/458 samples) are reported in the following table:

Table 10. ATLAS Temporal FiLM FV U-Net architecture ($C=392$, grid $R_f \times W_f=24 \times 42$).

Stage	Operation	Output	Spec.
Input	Frame / Frame+Video Attn.	$2C \times R_f \times W_f$	$784 \times 24 \times 42$
Encoder-1	3×3 Conv $\times 2$	$128 \times R_f \times W_f$	BN + ReLU
Downsample	2×2 MaxPool	$128 \times R_f/2 \times W_f/2$	–
Encoder-2	3×3 Conv $\times 2$	$256 \times R_f/2 \times W_f/2$	BN + ReLU
Bottleneck	3×3 Conv $\times 2$	$512 \times R_f/2 \times W_f/2$	–
Temporal	GAP $\rightarrow$ Transformer	512	2 layers, 4 heads
FiLM	γ, β	512	Channel-wise modulation
Upsample-2	Transposed Conv + skip (enc2)	$256 \times R_f/2 \times W_f/2$	Concat skip
Decoder-2	3×3 Conv $\times 2$	$256 \times R_f/2 \times W_f/2$	BN + ReLU
Upsample-1	Transposed Conv + skip (enc1)	$128 \times R_f \times W_f$	Concat skip
Decoder-1	3×3 Conv $\times 2$	$128 \times R_f \times W_f$	BN + ReLU
Output	1×1 Conv	$1 \times R_f \times W_f$	Segmentation logits

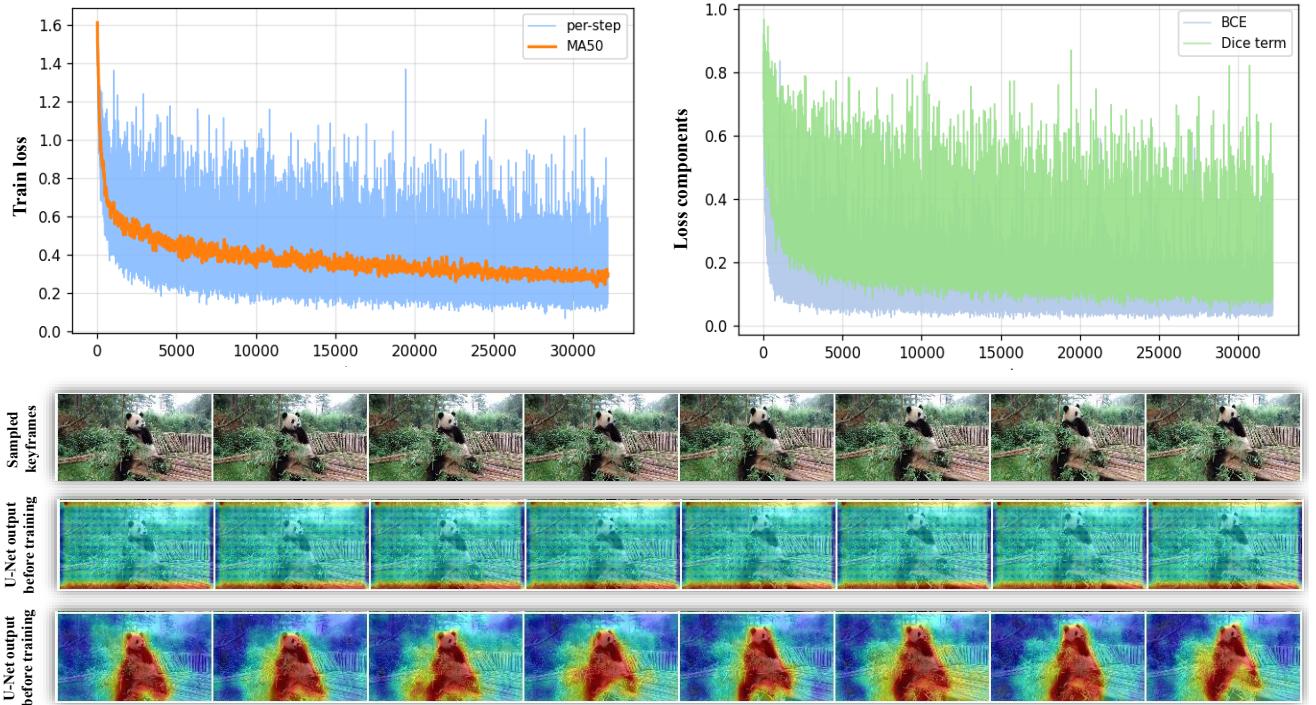


Figure 6. Training convergence of the U-Net adapter and qualitative outputs before and after training.

First failing stage	% of failures
Object extraction (LVLM)	33.6
Localization (adapter)	59.7
Tracklet selection	6.7
SAM2 propagation	0.0

Localization is the main bottleneck, accounting for 59.7% of failures, followed by object extraction at 33.6%. Here, object extraction refers to whether the LVLM correctly identifies the target category implied by the query be-

fore localization begins. Localization failures occur when the correct category is identified but the adapted attention map does not sufficiently isolate the referred object. Tracklet selection contributes only 6.7%, indicating that once a reliable localization map is available, candidate ranking is generally robust. SAM2 is never the first failing stage, suggesting that propagation errors typically originate from an incorrect or ambiguous seed rather than from SAM2 itself.

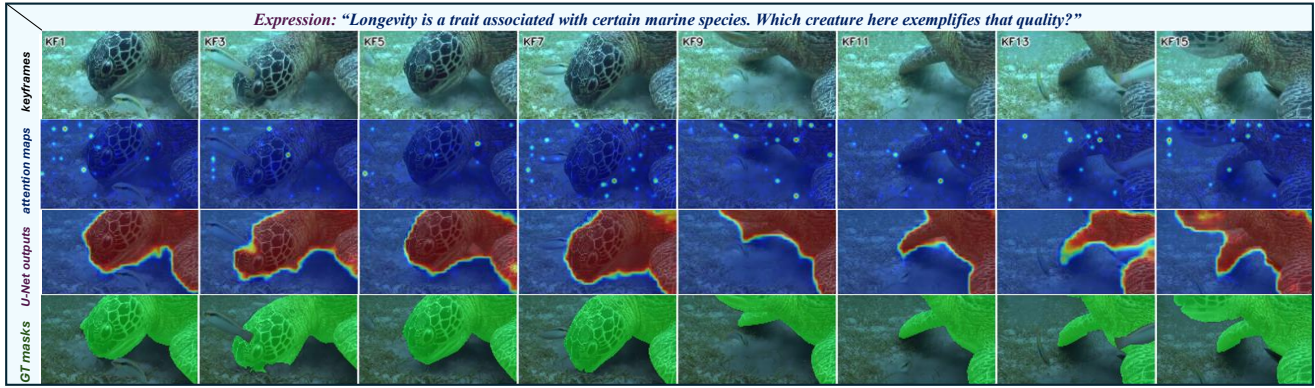


Figure 7. Additional qualitative example. From top to bottom: input frames, raw averaged attention maps, adapted attention maps produced by ATLAS, and the corresponding SAM2 segmentation masks.

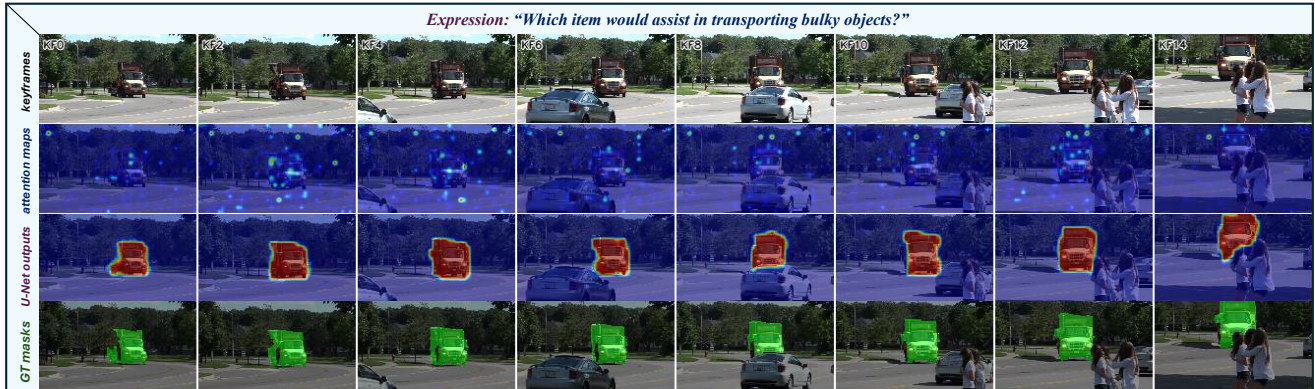


Figure 8. Additional qualitative example. From top to bottom: input frames, raw averaged attention maps, adapted attention maps produced by ATLAS, and the corresponding SAM2 segmentation masks.



Figure 9. Additional qualitative example. From top to bottom: input frames, raw averaged attention maps, adapted attention maps produced by ATLAS, and the corresponding SAM2 segmentation masks.

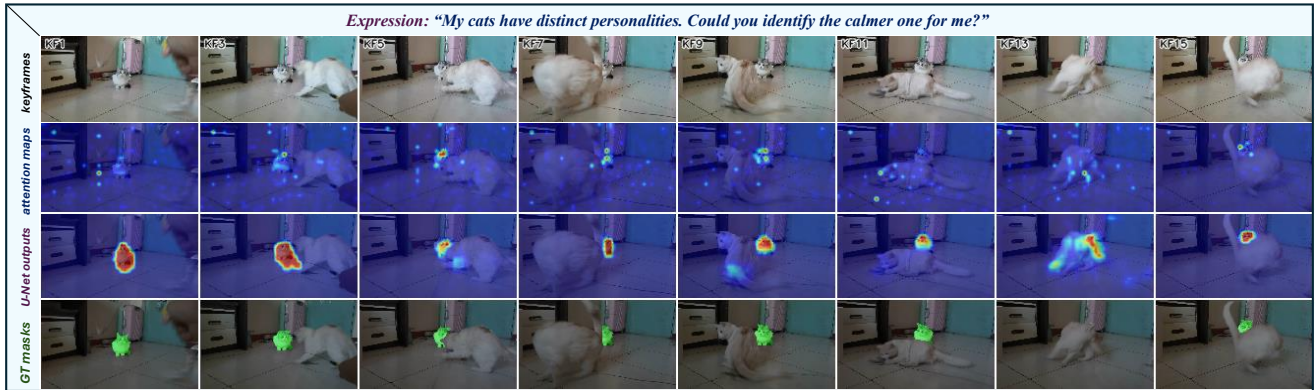


Figure 10. Additional qualitative example. From top to bottom: input frames, raw averaged attention maps, adapted attention maps produced by ATLAS, and the corresponding SAM2 segmentation masks.

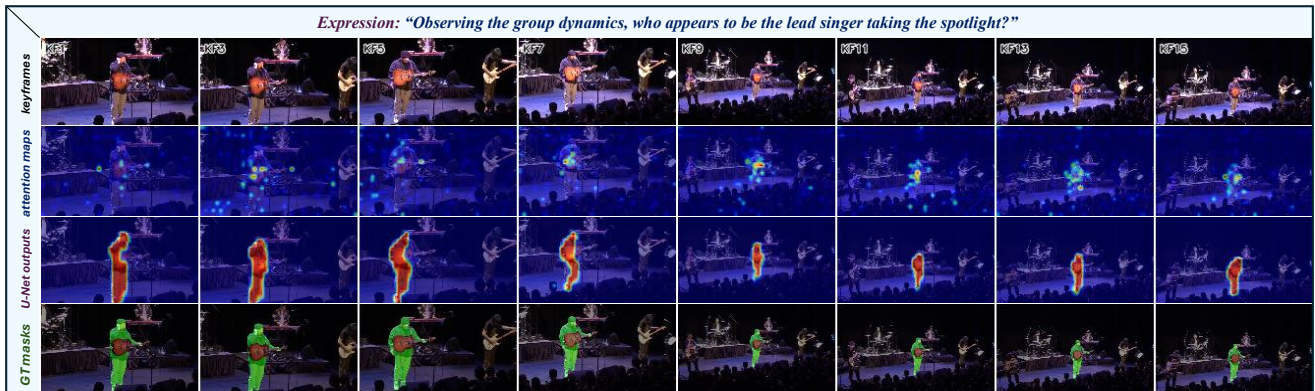


Figure 11. Additional qualitative example. From top to bottom: input frames, raw averaged attention maps, adapted attention maps produced by ATLAS, and the corresponding SAM2 segmentation masks.

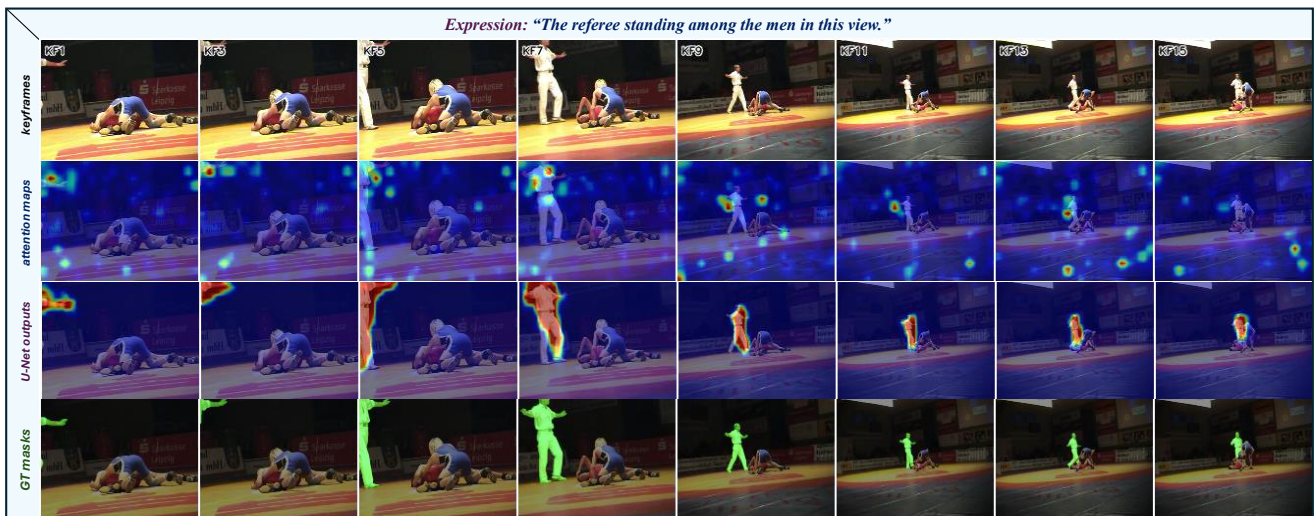


Figure 12. Additional qualitative example. From top to bottom: input frames, raw averaged attention maps, adapted attention maps produced by ATLAS, and the corresponding SAM2 segmentation masks.

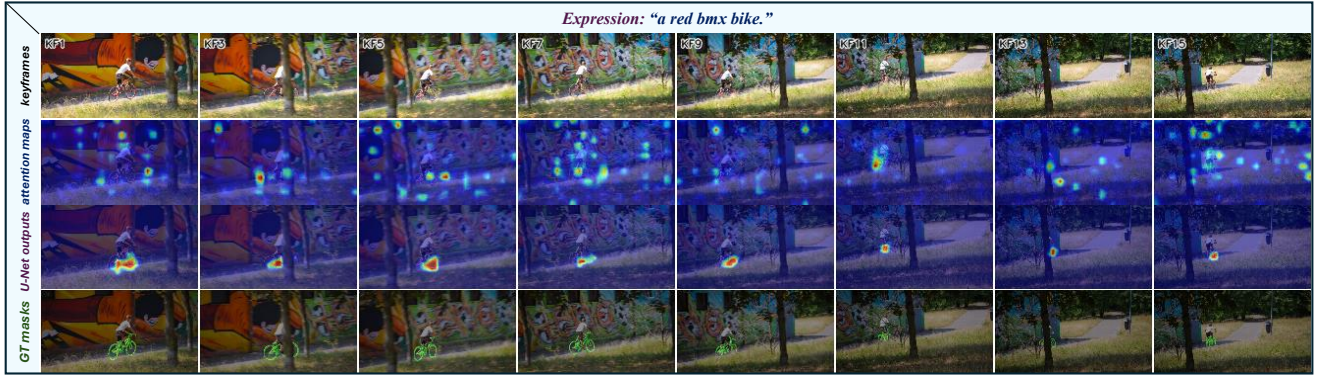


Figure 13. Additional qualitative example. From top to bottom: input frames, raw averaged attention maps, adapted attention maps produced by ATLAS, and the corresponding SAM2 segmentation masks.



Figure 14. Additional qualitative example. From top to bottom: input frames, raw averaged attention maps, adapted attention maps produced by ATLAS, and the corresponding SAM2 segmentation masks.

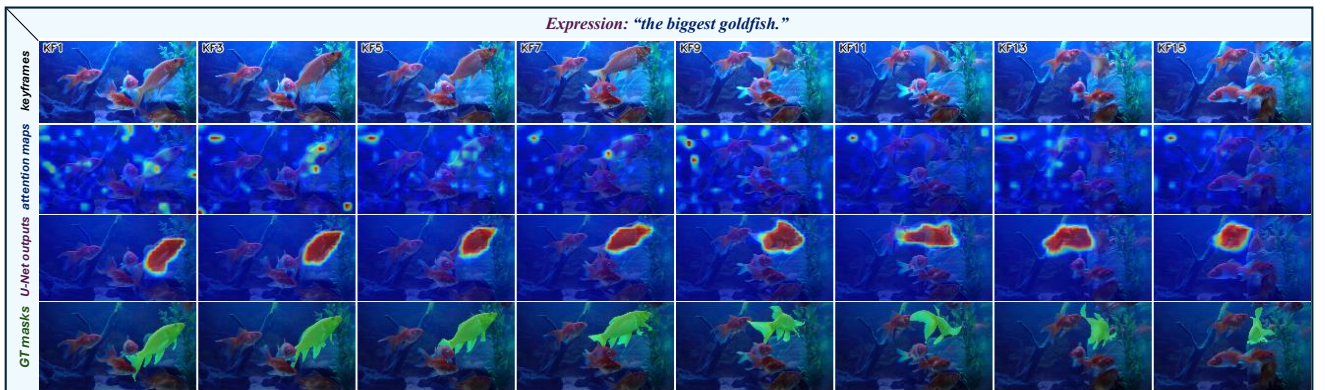


Figure 15. Additional qualitative example. From top to bottom: input frames, raw averaged attention maps, adapted attention maps produced by ATLAS, and the corresponding SAM2 segmentation masks.