Autoregressive Universal Video Segmentation Model

Appendix

A. Implementation Details

We further summarize key architectural configurations and experimental setups used throughout the training and evaluation of AUSM.

Model Configuration. The number of object queries for detection N^{det} and ID vectors N^{id} are both set to 100. The `History Compressor` module consists of 6 layers, and the following Transformer decoder contains 6 layers. For spatio-temporal fusion in `History Compressor`, we use the $1/8$ resolution feature map from the Swin backbone [27], with a fusion feature dimension of 256.

Our implementation does not use vision-language supervision (e.g., CLIP [36] or other pretrained image-text models); we instead employ dataset-specific classification heads. During training, each head is conditionally selected according to the dataset from which the input sample originates.

Training Setup. All experiments are conducted using 16 NVIDIA A100 GPUs with a batch size of 16 and an initial learning rate of 1×10^{-4}, optimized using AdamW [28] across all stages.

We begin the training process with pseudo-video training on the COCO dataset [26], using image instance segmentation pretrained Mask2Former [5] weights as initialization. This stage is trained for 20 epochs, corresponding to 147,500 iterations.

Next, we perform multi-source short-clip training using 5-frame video segments drawn from multiple datasets. This stage runs for 32,000 iterations and serves to adapt the model to short-term temporal dynamics and diverse visual domains.

Finally, we apply long-clip adaptation by fine-tuning the model on 16-frame clips for 40,000 iterations, enabling the model to better capture long-range temporal dependencies.

Inference Setup. Given a frame, we set the resolution of the shorter side to 1024. For Unprompted Video Segmentation, we apply a fixed foreground probability threshold of 0.5 to select confident detection predictions. Additionally, we perform top-k selection, retaining the top 10 instances per frame for YouTube-VIS 2019/2021 [52] and the top 20 for OVIS [34], to measure benchmark evaluation. These are the only post-processing steps employed; no other heuristics are introduced.

In contrast, prompted video segmentation involves no thresholding, filtering, or post-processing. Inference in this setting is fully model-driven, without reliance on manually designed components.

Implementation Details for Inference Scaling Experiments. We present additional methodological considerations for the scaling inference compute experiments. Due to the dense number of object appearing in the COCO validation set, we set dataset-specific foreground confidence thresholds to mitigate propagation of erroneous detections. Specifically, we use foreground confidence thresholds of 0.9 and 0.5 for COCO and YTVIS datasets, respectively. This stringent threshold for COCO results in a relatively modest performance metric (34.2 AP), as a considerable proportion of valid but lower-confidence detections are excluded from evaluation.

We further explore an alternative data augmentation strategy beyond simple image repetition. This approach involves strategic spatial decomposition of the input image into four overlapping quadrants: upper-left ($\mathcal{I}^{\text{UL}}$), upper-right ($\mathcal{I}^{\text{UR}}$), bottom-right ($\mathcal{I}^{\text{BR}}$), and bottom-left ($\mathcal{I}^{\text{BL}}$). Each quadrant maintains substantial spatial redundancy, with dimensions set to 90% of the original image size.

The resulting augmented sequence is formulated as:

$$\mathcal{I}^{\text{aug}} = (\mathcal{I}, \mathcal{I}^{\text{UL}}, \mathcal{I}^{\text{UR}}, \mathcal{I}^{\text{BR}}, \mathcal{I}^{\text{BL}}, \mathcal{I}) \tag{3}$$

This configuration enables the model to systematically traverse local regions while maintaining global context through the inclusion of the full image at sequence boundaries. Empirical evaluation demonstrates that this spatial traversal strategy yields statistically significant performance improvements, increasing mAP from 34.2 to 35.9 on the COCO validation set.

B. Qualitative Results

In Fig. 6 and Fig. 7, we present qualitative results to illustrate the capability of AUSM in both prompted and unprompted video segmentation settings. Notably, a single trained model is used for all results without any task-specific tuning or architectural changes. This underscores the unified nature of AUSM, which can seamlessly switch between prompted and unprompted modes at inference time.

For prompted segmentation, we visualize the model's ability to accurately segment target objects given masks in the keyframe. For unprompted segmentation, we show how the model discovers and tracks multiple objects throughout a video without external guidance. These results demonstrate that AUSM effectively handles both interaction-based and autonomous video understanding scenarios within a unified architecture.

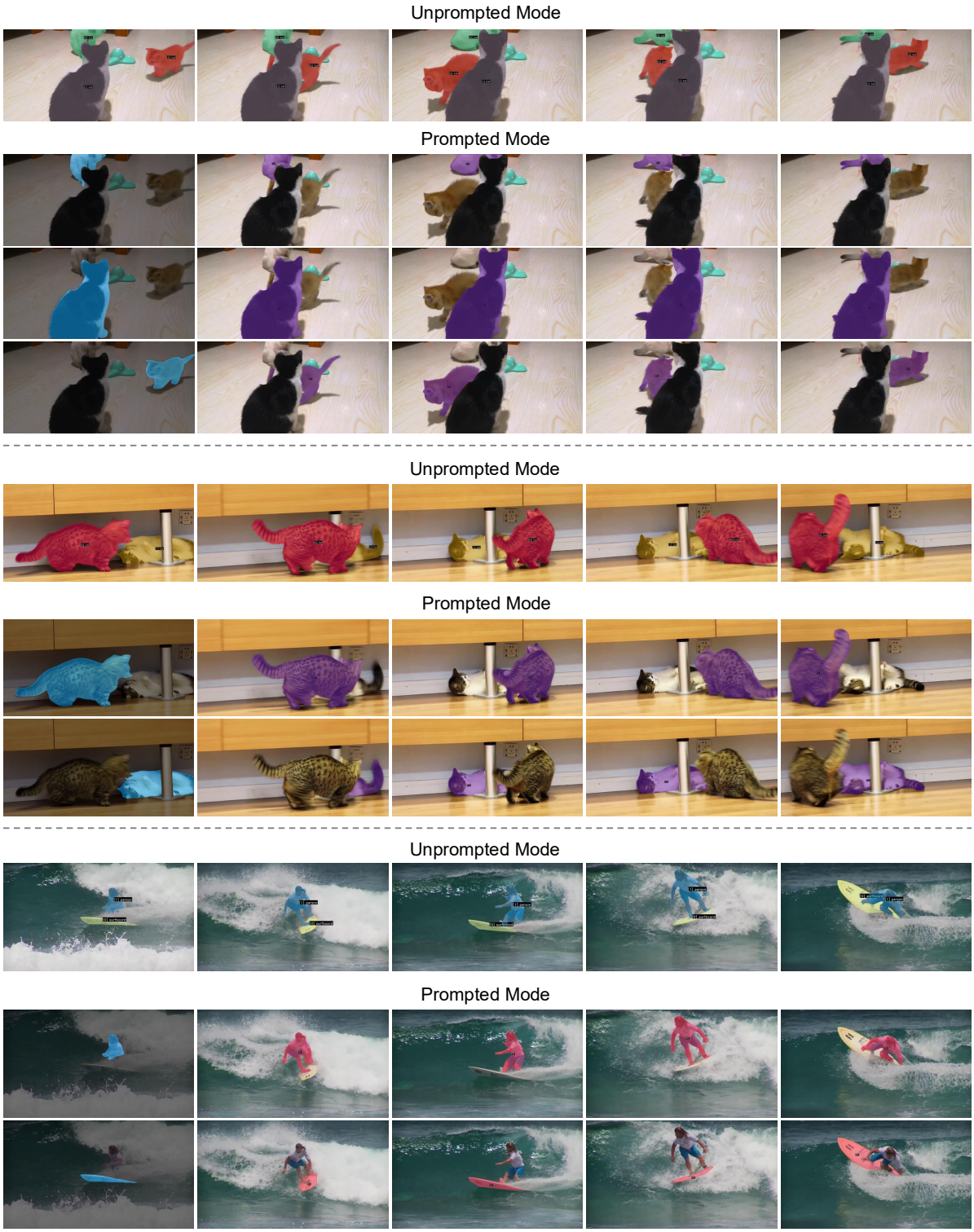

Figure 6. Qualitative comparison between unprompted and prompted video segmentation results using a single AUSM model. In the unprompted mode, the model autonomously discovers, segments, and classifies objects in the scene without any external guidance. In the prompted mode, it tracks only the object specified by an initial mask in the first frame. These examples demonstrate the unified capability of AUSM to seamlessly support both modes within a single framework.

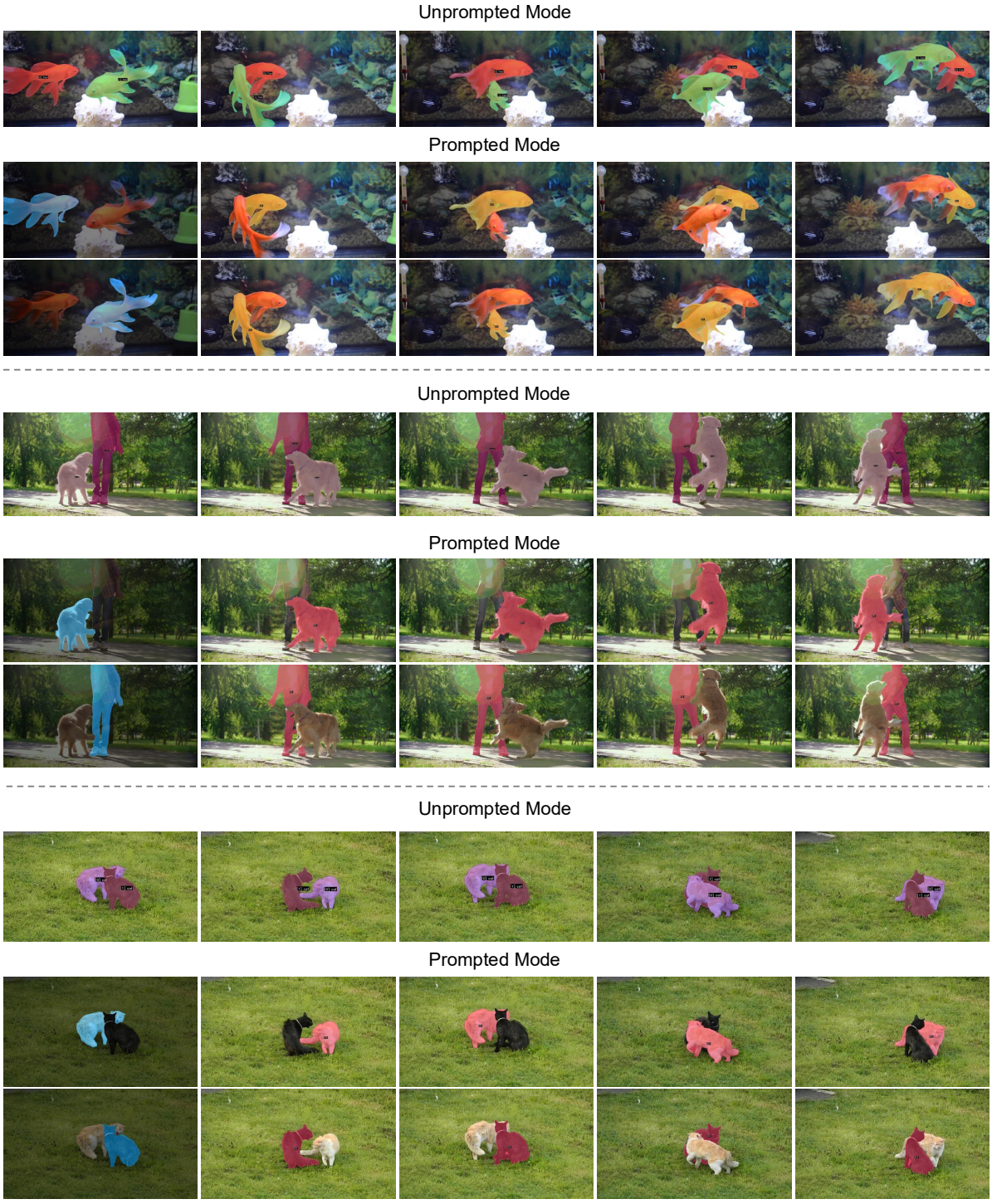

Figure 7. Qualitative comparison between unprompted and prompted video segmentation results using a single AUSM model. In the unprompted mode, the model autonomously discovers, segments, and classifies objects in the scene without any external guidance. In the prompted mode, it tracks only the object specified by an initial mask in the first frame. These examples demonstrate the unified capability of AUSM to seamlessly support both modes within a single framework.