Supplementary: Self-supervised co-salient object detection via feature correspondences at multiple scales

Souradeep Chakraborty[1] and Dimitris Samaras[1]

Department of Computer Science, Stony Brook University, USA
{souchakrabor,samaras}@cs.stonybrook.edu

In this document, we provide details about our experiments and present more results from our study. The document is organized into the following sections:

1. Section 1: Additional quantitative results
2. Section 2: Additional qualitative results
3. Section 3: Additional implementation details

1 Additional quantitative results

1.1 Performance using other encoder backbones

In Tab. 1, we show the effect of using different encoder backbones on the segmentation performance of our stage 1 self-supervised network. The ViT-Base encoder (embedding size = 768) with patch size = 8 provides the best performance, which we use in our final model. The convolution-based VGG-16 backbone [4] has a significantly worse performance compared to the other ViT-based backbones. For ViT encoders, we observe that increasing the patch size from 8 to 16 (or reducing the prediction resolution) leads to a significant drop in performance *e.g.* the F-measures on CoCA, Cosal2015 and CoSOD3k fall from 0.567, 0.844, 0.806 to 0.511, 0.760, and 0.664 respectively. Also, reducing the encoder's representation ability using a reduced embedding size (using the ViT-Small backbone) while keeping the patch size same leads to a drop in performance *e.g.* the F-measures on CoCA, Cosal2015 and CoSOD3k fall from 0.567, 0.844, 0.806 to 0.560, 0.830, and 0.752 respectively.

Table 1: The effect of using different encoder backbones on the segmentation performance of our stage 1 self-supervised network. The ViT-Base encoder with patch size = 8 provides the best performance.

			CoCA [9]				Cosal2015 [8]				CoSOD3k [2]			
Encoder	Patch size	Embedding size	MAE↓	F_β^{max} ↑	E_ϕ^{max} ↑	S_α ↑	MAE↓	F_β^{max} ↑	E_ϕ^{max} ↑	S_α ↑	MAE↓	F_β^{max} ↑	E_ϕ^{max} ↑	S_α ↑
VGG-16	16	512	0.115	0.356	0.632	0.518	0.205	0.475	0.553	0.505	0.173	0.468	0.572	0.517
ViT-Base	16	768	0.116	0.511	0.743	0.640	0.092	0.760	0.863	0.785	0.119	0.664	0.792	0.724
ViT-Small	8	384	0.105	0.559	0.755	0.667	0.091	0.810	0.851	0.789	0.081	0.779	0.852	0.778
ViT-Base	8	768	0.104	0.567	0.756	0.679	0.069	0.844	0.894	0.832	0.069	0.806	0.878	0.808

Based on the ablation study in Tab. 2 we set the similarity threshold d_f^{th} to 0.75. Increasing or decreasing this value produced inferior performance.

Table 2: Performance comparison of our SCoSPARC using different values of d_f^{th}.

	CoCA				CoSal2015				CoSOD3k			
d_f^{th}	$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\phi^{max} \uparrow$	$S_\alpha \uparrow$	$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\phi^{max} \uparrow$	$S_\alpha \uparrow$	$MAE \downarrow$	$F_\beta^{max} \uparrow$	$E_\phi^{max} \uparrow$	$S_\alpha \uparrow$
0.50	0.096	0.588	0.786	0.702	0.065	0.861	0.903	0.850	0.065	0.824	0.891	0.823
0.75	**0.081**	**0.637**	**0.814**	**0.738**	**0.056**	**0.891**	**0.924**	**0.881**	**0.062**	**0.834**	**0.901**	**0.843**
0.90	0.087	0.620	0.803	0.718	0.066	0.866	0.903	0.848	0.067	0.826	0.885	0.816

2 Additional qualitative results

2.1 Comparison of CoSOD predictions

In Fig. 1 we qualitatively compare the CoSOD predictions from our SCoSPARC model with two unsupervised CoSOD models US-CoSOD [1] and Group TokenCut and with four recent supervised models CoRP [6], DCFM [7], UFO [5] and GCoNet+ [10].

For the *Calculator* class, we observe that the US-CoSOD model produces undesirable image regions as CoSOD detections. Our Group TokenCut baseline produces reasonably good detections in this case, although there are edge artifacts. The supervised models such as CoRP and DCFM produce incomplete segmentations in several instances (columns 1, 3 and 4). The DCFM model also segments undesirable image regions *e.g.* the paper in column 1 and the pen in column 2. The UFO model inaccurately segments the pen as being co-salient in column 3. Finally, the GCoNet+ model, although being SOTA in supervised CoSOD, produces noisy predictions for this image group. Our model produces the best results in general.

For the *Coffecup* class, in column 1 we observe that all models except SCoSPARC produce either incomplete segmentations (*e.g.* not detecting the textual regions on the cup) or inaccurately segment undesirable regions (*e.g.* CoRP segments the background region between the cup handle). Similarly, in column 2, most baseline models inaccurately segment the background region between the cup handle. Also, in the third column, we see that our model produces the best results while other baselines produce inaccurate segmentations. In column 4, Group TokenCut produces comparable results to our predictions, while other models produce noisy segmentations.

For the *Bee* class, most models except Group TokenCut inaccurately detect flower parts as being co-salient. While Group TokenCut produces reasonable segmentations in this case, the predictions of our SCoSPARC are more refined.

2.2 Visualizations of intermediate maps

In Fig. 2 we show additional visualizations of the intermediate self-attention maps, cross-attention maps and segmentation maps for two image groups, *Hat* and *Accordion* (three instances each) from the CoCA dataset. The yellow boxes highlight the regions eliminated using the stage 2 of our SCoSPARC model. We

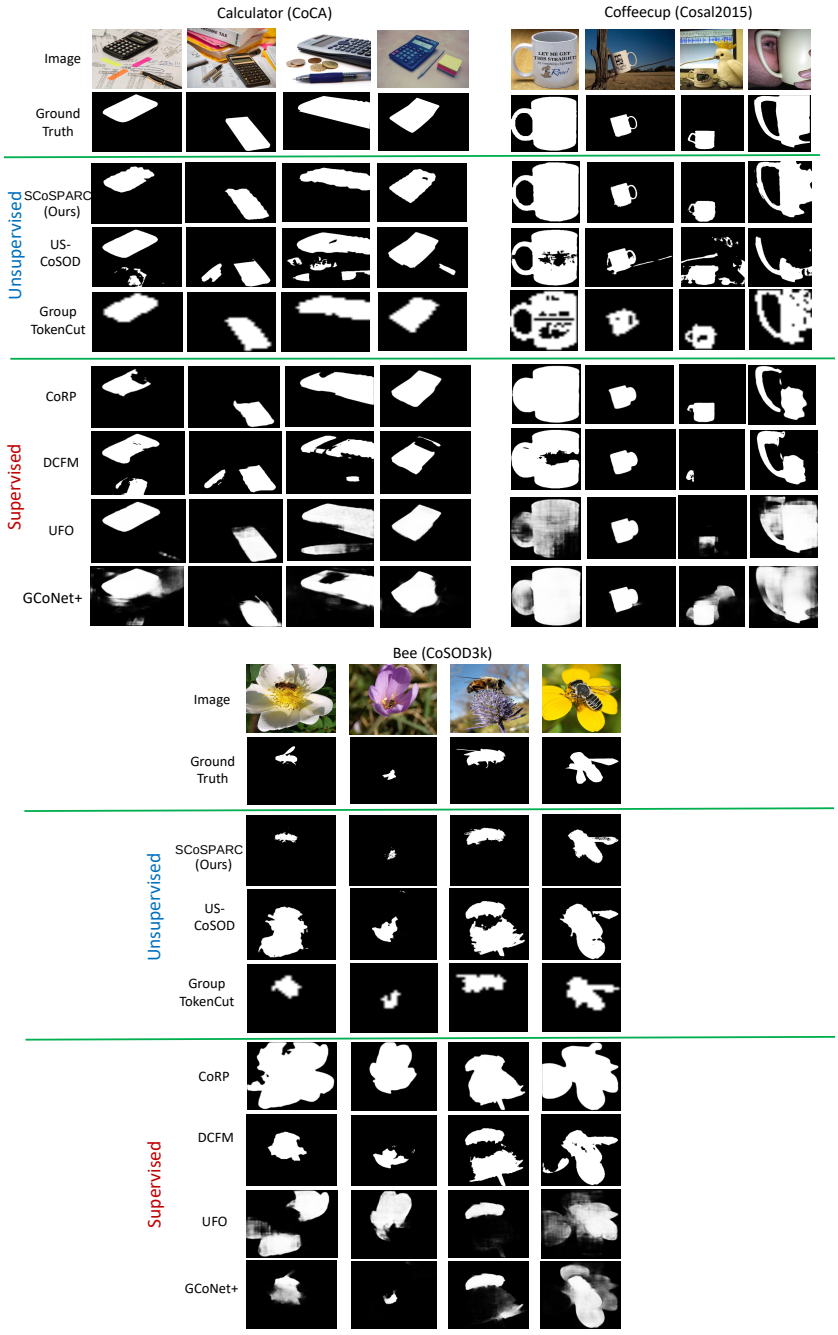

Fig. 1: Additional qualitative comparison of the performance of different baselines with our self-supervised CoSOD model on three image groups, each selected from the CoCA, Cosal2015, and CoSOD3k datasets. Our model produces the most accurate segmentations.

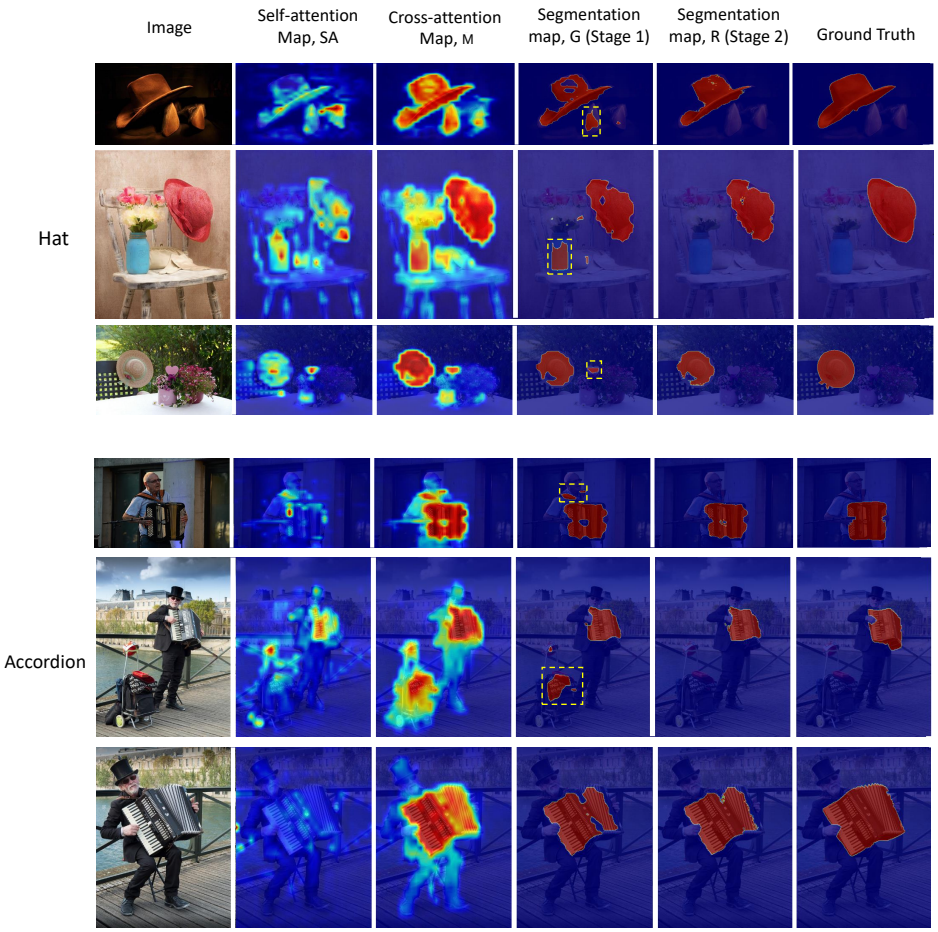

Fig. 2: Additional visualizations of the intermediate self-attention maps, cross-attention maps and segmentation maps for two instances from the *handbag* category from the CoCA dataset. The yellow boxes highlight the regions eliminated using the stage 2 of our SCoSPARC model.

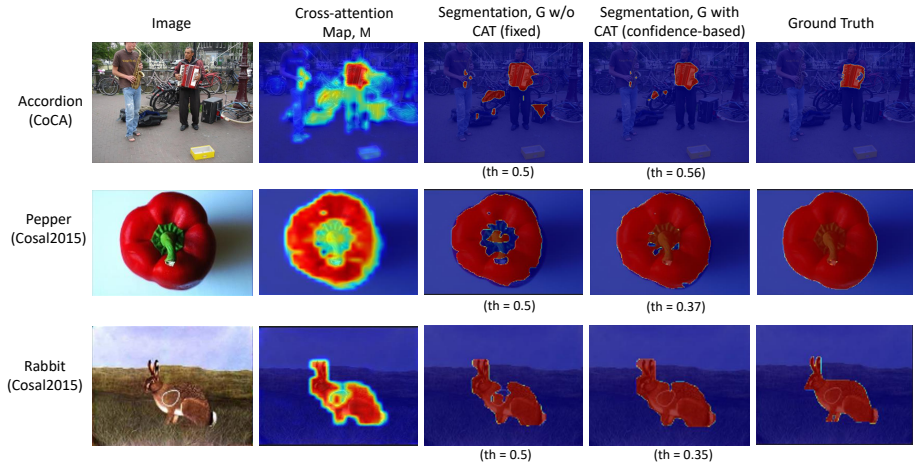

Fig. 3: Visualizations of the segmentation map, G (from stage 1) with and without our confidence-based adaptive thresholding (CAT) component. Our model with CAT produces more accurate segmentation predictions.

observe that undesirable image regions (i.e. non-co-salient regions highlighted by the yellow boxes) are eliminated in stage 2 segmentation predictions, $\mathcal{R}$ from our model using our region-level feature correspondence step (RFC).

2.3 Visualizations of confidence-based adaptive thresholding results

In Fig. 3, we visualize the segmentation maps, G (from stage 1) with and without our confidence-based adaptive thresholding (CAT) component. We see that our model with CAT produces more accurate segmentation predictions. In row 1 of Fig. 3 (for an instance from the *Accordion* category), we see that the CAT step eliminates undesirable image regions using a higher threshold of 0.56 (determined via prediction confidence) compared to the segmentation obtained using a fixed threshold of 0.5 (widely used in segmentation tasks). On the other hand, for the categories *Pepper* and *Rabbit*, we see that lower threshold values of 0.37 and 0.35 produces better segmentations respectively, compared to the fixed 0.5 threshold. The different threshold values predicted by our CAT step are based on the different average confidence intensities of the confident regions in the cross-attention map, $\mathcal{M}$ for the three cases. For example, the per-pixel confidence value (within the confident regions) of the map S for the *Accordion* category (row 1) is lesser than the per-pixel confidence values for the *Pepper* and *Rabbit* categories, which leads the algorithm to predict a higher threshold for *Accordion* compared to the other two categories in rows 2 and 3. This results in improved segmentations.

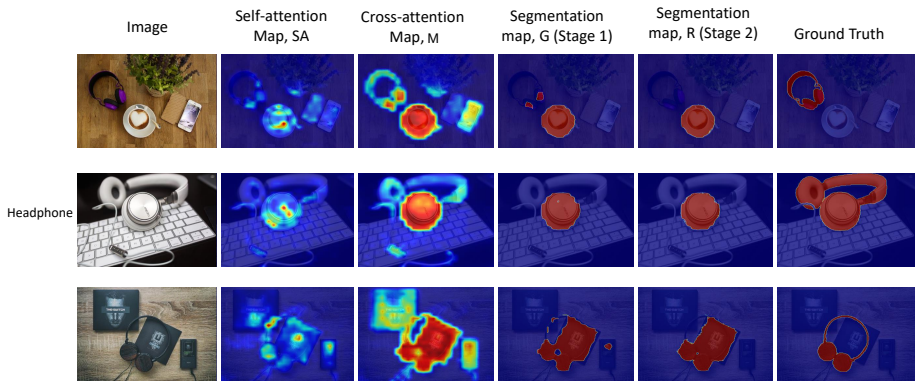

Fig. 4: Some failure cases of predictions on the *headphone* image group from CoCA.

2.4 Failure cases

In Fig. 4 we show some failure cases of our SCoSPARC model on the *Headphone* image group from the CoCA dataset. In row 1 the model misses the headphone and instead highlights the cup and plate as the co-salient objects. In row 2 only one side of the headphone object has been accurately segmented while the model fails to detect the other half including the headband. We observe that a lower threshold on the cross-attention map, $\mathcal{M}$ could have produced an improved segmentation (highlighting all parts of the headphone), which our model fails to predict. In row 3, our model predicts certain undesirable regions as being co-salient along with the headphone.

3 Additional implementation details

3.1 Training details

We use the ViT-Base model (with patch size = 8 and patch descriptor dimension = 768) trained using DINO as our backbone feature extractor. We freeze the weights of this backbone for all of our experiments. See Tab. 1 for more ablations on the encoder choice. For training, we set the sample size as the minimum of 24 or the total group size. We input images with size 224×224. Using the ViT-Base model with patch size = 8 produces co-attention maps with size $(\frac{224}{8}, \frac{224}{8}) = 28 \times 28$.

We used the PyTorch deep learning library and the Adam optimizer to train our stage 1 network. We set the learning rate to 10^{-4} and the weight decay parameter to 10^{-4}. The total training time for SCoSPARC is around 10 hours for 80 epochs. All experiments are run on an NVIDIA Quadro RTX 8000 GPU.

3.2 Inference details

At inference, all samples (resized to 224×224) in the group are input at once. The inference speed of the model is 20.5 FPS (without DenseCRF) and 4.1 FPS (with DenseCRF).

For the dense CRF [3] post-processing step , we generated the unary operator directly from the binary segmentation map, $\mathcal{R}$ from stage 2. We set the smoothness kernel parameter $\theta_\gamma = 10$ and the appearance kernel parameters, θ_α and θ_β to 10 and 3 respectively.

References

1. Chakraborty, S., Naha, S., Bastan, M., C., A.K.K., Samaras, D.: Unsupervised and semi-supervised co-salient object detection via segmentation frequency statistics. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 332–342 (January 2024) 2
2. Fan, D.P., Lin, Z., Ji, G.P., Zhang, D., Fu, H., Cheng, M.M.: Taking a deeper look at co-salient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2919–2929 (2020) 1
3. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. Advances in neural information processing systems **24** (2011) 7
4. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) 1
5. Su, Y., Deng, J., Sun, R., Lin, G., Su, H., Wu, Q.: A unified transformer framework for group-based segmentation: Co-segmentation, co-saliency detection and video salient object detection. IEEE Transactions on Multimedia (2023) 2
6. Wu, Y., Song, H., Liu, B., Zhang, K., Liu, D.: Co-salient object detection with uncertainty-aware group exchange-masking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19639–19648 (2023) 2
7. Yu, S., Xiao, J., Zhang, B., Lim, E.G.: Democracy does matter: Comprehensive feature mining for co-salient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 979–988 (2022) 2
8. Zhang, D., Han, J., Li, C., Wang, J., Li, X.: Detection of co-salient objects by looking deep and wide. International Journal of Computer Vision **120**(2), 215–232 (2016) 1
9. Zhang, Z., Jin, W., Xu, J., Cheng, M.M.: Gradient-induced co-saliency detection. In: European Conference on Computer Vision. pp. 455–472. Springer (2020) 1
10. Zheng, P., Fu, H., Fan, D.P., Fan, Q., Qin, J., Tai, Y.W., Tang, C.K., Van Gool, L.: Gconet+: A stronger group collaborative co-salient object detector. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023) 2