RADSeg: Unleashing Parameter and Compute Efficient Zero-Shot Open-Vocabulary Segmentation Using Agglomerative Models

Supplementary Material

Limitations

While **RADSeg**-base delivers strong mIoU gains with a lightweight vision encoder, it relies on a huge text encoder, unlike CLIP-based baselines in the same model size class. However, text features are computed once per query, so their cost is amortized across many images which. Furthermore, although we demonstrate **RADSeg**'s strong performance on eight 2D and 3D OVSS datasets, current OVSS benchmarks remain limited. They primarily test whether each pixel is more similar to query A, B, **or** C. A more challenging setting would prevent access to all class labels and evaluate queries individually (is this pixel similar to query A?). Moreover, support for multi-label segmentation is not explored. Developing and evaluating on such challenging settings is left for future work.

Acknowledgments

This work was supported by Defense Science and Technology Agency (DSTA) Contract #DST000EC124000205, King Abdulaziz University, National Institute on Disability, Independent Living, and Rehabilitation Research (NIDILRR) Grant #90IFDV0042, and DENSO Corporation Grant #OSP00016426. The work additionally used Bridges-2 at PSC through allocation cis220039p from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program which is supported by NSF grants #2138259, #2138286, #2138307, #2137603, and #213296. We also gratefully acknowledge NVIDIA for providing GPUs to Airlab through academic hardware grants. Finally, we thank Mike Ranzinger, Nikhil Keetha, and Parv Maheshwari for insightful discussions.

A1. Contribution Statement

Omar Alama Led and shaped the research, conceived the initial ideas for the encoder, wrote the majority of the manuscript, designed key figures and table formats. Additionally, developed the 3D evaluation pipeline and provided coding support and debugging throughout the project.

Darshil Jariwala Developed the encoder code and refined the initial ideas, conducted all evaluations and ablations for 2D OVSS – including porting and optimizing 2D baselines, generated qualitative 2D results, and helped with paper writing.

Avigyan Bhattacharya Conducted all evaluations for 3D OVSS, including porting 3D baselines, generated 3D qualitative figures, adapted ResCLIP to RADIO, wrote the 2D and 3D experimental sections. Contributed to paper writing and refinement.

Seungchan Kim Incorporated and processed the ScanNet++ dataset for 3D OVSS evaluation, and played a key role in refining the manuscript, improving clarity, structure, and presentation.

Wenshan Wang Provided valuable feedback on research design, manuscript writing, and method presentation.

Sebastian Scherer Provided valuable feedback on research direction, manuscript clarity, and figure presentation.

A2. Additional 2D Evaluation Details

Table A.1. Adopted resolution and sliding window settings. Cell format: Shorter side - Crop - Stride. TextRegion's stride always equals the crop size as per their method.

Dataset	Low [13, 39, 44]	Mid [36]	High [42]
VOC	336-224-112	336-336-112	672-336-336
Stuff	336-224-112	448-336-224	896-336-336
CTX/ADE	336-224-112	576-336-224	672-336-336
City	560-224-112	688-336-224	1344-336-336

Emphasis on resolution standardization. By examining the settings used for evaluation in previous works, we observe varying resolution standards, not only across methods and datasets, but even within different sub-modules of a method itself. These inconsistencies can conflate performance gains from increased resolutions with method improvements. For example, while NACLIP [13] and ResCLIP [44] choose the low resolution setting in their evaluations, Tab. 1 shows that they benefit from increased resolutions at the mid setting, making it unfair to compare their low-resolution performance with mid or high resolution performances of other approaches, which is done in many previous works [36, 42]. Furthermore, previous works have shown that increasing resolution can hurt performance [4, 36]. Thus, for a fair standard comparison, we rerun all baselines with 3 different representative resolution and sliding window standards (low, mid, high) shown in Tab. A.1. Note that evaluation resolution, at which we compute metrics, always remains at the ground truth resolution. To avoid penalizing approaches that perform better when using lower resolutions, **we report the maximum performance at a resolution limit** in Tab. 1. Tab. 1 answers what the best performance is for a method given a **resolution limit**, while its source data, shown in Tab. A.2, answers

Table A.2. 2D zero-shot open-vocabulary semantic segmentation mIoU results across resolutions and model sizes. Ranking shown as **first**, second, and third. One ranking for base and huge is reported to highlight that **RADSeg**-base is also superior to huge baselines.

Methods	= Low Resolution						= Mid Resolution						= High Resolution					
	CTX	VOC	Stuff	ADE	City	Avg	CTX	VOC	Stuff	ADE	City	Avg	CTX	VOC	Stuff	ADE	City	Avg
Base Models																		
NACLIP [13]	35.17	79.71	23.3	17.42	35.49	38.22	34.7	80.29	23.64	17.78	35.98	38.48	33.86	72.57	20.04	17.8	36.74	36.20
ResCLIP [44]	36.8	82.32	24.7	18.03	35.85	39.54	36.75	85.89	25.07	18.55	36.19	40.49	35.98	76.6	22.01	18.57	36.88	38.01
RayFronts [1]	36.81	72.59	25.34	23.38	36.29	38.88	38.07	80.12	27.39	24.63	40.47	42.14	36.99	67.39	23.92	24.18	39.13	38.32
ProxyCLIP [20]	38.74	78.18	26.11	19.71	39.69	40.49	37.88	80.31	26.41	19.67	40.39	40.93	34.85	70.44	21.63	19.18	38.84	36.99
SC-CLIP [3]	40.12	84.29	26.62	20.06	41.02	42.42	39.87	87.67	27.25	20.68	40.49	43.19	38.68	77.67	22.89	20.34	41.24	40.16
Talk2Dino [4]	39.43	85.68	27.43	20.42	38.05	42.20	40.31	85.66	27.89	21.67	39.47	43.00	40.23	84.15	27.06	21.70	41.15	42.86
Trident [36]	41.62	83.74	28.22	21.17	41.86	43.32	42.16	84.5	28.24	21.98	42.08	43.79	40.70	80.75	25.45	20.81	42.19	41.98
TextRegion [42]	41.53	83.83	27.39	21.21	40.49	43.17	42.29	84.21	27.13	21.74	41.26	43.33	42.88	83.19	28.62	22.55	42.15	43.88
RADSeg	**44.49**	**87.24**	**29.79**	**27.16**	**42.04**	**46.14**	**45.64**	**89.28**	**30.76**	**28.96**	**45.35**	**48.00**	**45.14**	**84.07**	**29.05**	**28.93**	**48.79**	**47.20**
RADSeg+	**48.24**	**88.46**	**31.69**	**29.63**	**45.58**	**48.84**	**48.48**	**90.14**	**32.48**	**30.83**	**48.10**	**50.01**	**47.86**	**85.70**	**30.43**	**30.86**	**50.96**	**49.49**
Huge Models																		
RayFronts [1]	31.67	72.59	23.31	20.46	28.98	35.40	32.29	73.36	23.44	21.19	31.84	36.42	32.19	67.64	22.13	21.08	34.27	35.46
ProxyCLIP [20]	39.16	78.02	26.19	23.90	43.64	42.18	38.31	83.03	27.76	24.05	43.92	43.21	37.03	71.66	21.81	23.65	43.09	39.45
Trident [36]	43.16	87.97	28.55	25.64	46.87	46.44	44.32	88.67	28.52	26.70	46.30	46.90	43.77	87.22	25.72	27.02	47.34	46.21
TextRegion [42]	44.04	89.53	30.19	24.53	47.35	47.13	44.76	89.42	29.85	26.42	47.04	47.50	46.13	89.36	31.22	27.30	46.88	48.18
RADSeg	**42.27**	**89.58**	**28.3**	**25.96**	**38.85**	**44.99**	**44.8**	**89.74**	**28.93**	**28.21**	**42.93**	**46.92**	**45.01**	**88.52**	**29.46**	**27.15**	**47.75**	**47.58**
RADSeg+	**45.75**	**90.39**	**30.17**	**27.86**	**42.49**	**47.73**	**47.76**	**90.44**	**30.50**	**30.09**	**46.48**	**49.38**	**47.98**	**89.40**	**30.79**	**29.99**	**50.58**	**50.19**

what the performance is at a **particular resolution**. While Tab. 1 is more relevant for method comparisons, we report Tab. A.2 for completeness. Note that **all other** ablations and tables use mid resolution only.

A3. Additional 3D Evaluation Details

3D mapping for evaluation. We follow RayFronts' [1] approach to construct the 3D map and perform the evaluation. Given a pose $P_t \in SE(3)$, a corresponding depth map $D_t \in \mathbb{R}^{H\times W}$, and a feature map $F_t \in \mathbb{R}^{H\times W\times D}$, feature pixels are back projected to a 3D point cloud $P_t^{\text{local}} = \{(p_i, f_i)\}_{i=1}^{M}$, where $p_i \in \mathbb{R}^3$ denotes 3D position, and $f_i \in \mathbb{R}^{D+1}$ represents the concatenated feature vector, and hit count (initialized to 1 per point). Local updates are accumulated over frames and voxelized at a resolution of α to form the global semantic voxel map $P_t^{\text{global}} = \{(p_i, f_i)\}_{i=1}^{N}$. During voxelization, points that lie in the same voxel get their features averaged, and their hit counts summed to use as weights for subsequent averaging. In case of probability space 3D aggregation, the embedding dimension D in the feature map and semantic voxel map is equal to the number of classes as segmentation probabilities are projected instead of embeddings.

Datasets. Following prior works [1, 12, 15, 40] for our 3D OVSS evaluation, we choose the scenes, `office[0-4]`, `room[0-2]` from Replica and `scene[0011,0050,0231,0378,0518]` from ScanNet. Additionally, to assess performance on a cleaner real-world dataset, we evaluate on ScanNet++ and select nine diverse scenes. The selected scenes are `scene[00777c41d4, bcd2436daf, a5114ca13d, 2b1dc6d6a5, d551dac194, f9f95681fd, ea42cd27e6, 20ff72df6e, bf6e439e38]`. We use all 101 classes from Replica, the standard ScanNet-to-NYU40 label mapping provided with the dataset itself for ScanNet, and the top 100 classes from ScanNet++ as defined in its official semantic segmentation benchmark. In ScanNet, we assign three of the forty classes ('otherprop,' 'otherstructure,' and 'otherfurniture') as ignore classes due to their ambiguity.

Implementation details. For all the datasets, we resize each image by setting the shorter side to 640 and keeping the original aspect ratio. A frame skip of 10 and 5cm voxels are used for constructing the 3D map. We use external ground-truth voxels for Replica (following [1], we use those provided by HOV-SG) and for ScanNet++. For ScanNet, however, we derive the ground truth by lifting the 2D semantic segmentation annotations into 3D. During evaluation, we perform k-NN matching (k = 5) in accordance with the HOV-SG [40] protocol, assigning each ground-truth voxel the majority label of its nearest neighbors.

A4. Additional Efficiency Evaluation Details

Improving baselines efficiency. To have a robust comparison, we modify the inference code of NACLIP [13], ResCLIP [44], SC-CLIP [3], and ProxyCLIP [20] to use batched sliding window inference as opposed to iterating over each window. This significantly improves the baselines latency and gives us stronger comparison points. In our experiments, all baselines use batched sliding windows with a batch size equal to the number of windows.

What latency to report? Input resolutions vary across datasets and even across samples since only the shorter image side is fixed. Some methods adjust hyperparameters

Table A.3. Compute and parameter efficiency across datasets on a V100 GPU at mid resolution and FP32 precision. The table reports vision encoder parameters, average mIoU across datasets, and per-dataset latency to capture differences arising from varying input resolutions—both across and within datasets. Illustrated visually in Fig. 1. At mid resolution, **RADSeg-base surpasses huge Trident and TextRegion baselines while being 9.3x-12.6x faster and having 8.1x-12.7x fewer parameters**.

Methods	Latency (s)					Lat (s)	Params (M)	mIoU (%)
	CTX	VOC	Stuff	ADE	City	Avg		
Base Models								
NACLIP [13]	0.091	0.025	0.074	0.111	0.144	0.089	86	38.48
ResCLIP [44]	0.252	0.152	0.266	0.325	0.311	0.261	86	40.33
RayFronts [1]	0.106	0.035	0.087	0.109	0.218	0.111	103	42.14
ProxyCLIP [20]	0.689	0.262	0.247	0.693	1.446	0.667	171	40.87
SC-CLIP [3]	0.148	0.059	0.111	0.174	0.243	0.147	86	41.86
Trident [36]	0.353	0.229	0.348	0.414	0.923	0.454	264	43.79
TextRegion [42]	0.410	0.403	0.438	0.428	1.729	0.682	167	43.33
Talk2Dino [4]	0.150	0.043	0.155	0.188	0.289	0.165	86	43.00
RADSeg	**0.109**	**0.036**	**0.088**	**0.112**	**0.228**	**0.115**	**106**	**48.00**
RADSeg+	**0.261**	**0.156**	**0.286**	**0.360**	**0.709**	**0.354**	**119**	**50.01**
Huge Models								
RayFronts [1]	0.619	0.204	0.460	0.598	1.362	0.649	661	36.40
ProxyCLIP [20]	1.297	0.494	0.451	1.28	3.168	1.338	717	43.21
Trident [36]	1.426	0.900	1.123	1.444	2.365	1.451	1349	46.90
TextRegion [42]	0.732	0.670	0.711	0.744	2.476	1.067	857	47.50
RADSeg	**0.617**	**0.202**	**0.457**	**0.594**	**1.363**	**0.647**	**664**	**46.92**
RADSeg+	**1.172**	**0.734**	**1.074**	**1.341**	**2.128**	**1.290**	**677**	**49.38**

per dataset, affecting total computation. Furthermore, the latency of methods (including **RADSeg+**) can vary with the content of an image as the number of masks varies. To provide a holistic measure that reflects all computations contributing to a method's final segmentation accuracy, we report the average latency across all validation samples in each dataset, as shown in Tab. A.3. The table reveals substantial latency differences between datasets, underscoring that measuring latency on a single image or at a fixed resolution does not capture overall performance.

Latencies are measured using mid resolution, FP32 precision, on a Tesla V100-32GB GPU. In addition, the number of parameters of the vision encoders of each method is reported. Fig. 1 visualizes the mIoU vs number of parameters and mIoU vs latency tradeoffs. Notably, at mid resolution, **RADSeg-base surpasses huge Trident and TextRegion baselines while being 9.3x-12.6x faster and having 8.1x-12.7x fewer parameters**.

Table A.4. **SCRA+SCGA generalize** consistently across all RADIO versions and sizes improving mIoU. Rv3-l is excluded due to lack of CLS-patch alignment.

Version	Method	CTX	VOC	Stuff	ADE	City	Avg
Rv2.5-b	Base	34.24	84.99	24.66	23.11	32.78	39.96
	+SCRA+SCGA	44.52	87.59	30.52	28.27	43.56	46.89
	Δ	**+10.28**	**+2.60**	**+5.86**	**+5.16**	**+10.78**	**+6.93**
Rv2.5-l	Base	31.70	83.88	23.68	21.00	32.49	38.55
	+SCRA+SCGA	38.85	89.02	28.14	25.53	43.72	45.05
	Δ	**+7.15**	**+5.14**	**+4.46**	**+4.53**	**+11.23**	**+6.50**
Rv2.5-h	Base	31.12	79.04	23.00	21.92	29.78	36.97
	+SCRA+SCGA	38.67	82.52	27.31	25.81	36.63	42.19
	Δ	**+7.55**	**+3.48**	**+4.31**	**+3.89**	**+6.85**	**+5.22**
Rv3-b	Base	40.40	88.07	27.41	27.30	39.94	44.62
	+SCRA+SCGA	45.64	89.28	30.76	28.96	45.35	48.00
	Δ	**+5.24**	**+1.21**	**+3.35**	**+1.66**	**+5.41**	**+3.38**
Rv3-h	Base	35.23	88.35	24.55	24.34	33.74	41.24
	+SCRA+SCGA	44.80	89.74	28.93	28.21	42.93	46.92
	Δ	**+9.57**	**+1.39**	**+4.38**	**+3.87**	**+9.19**	**+5.68**

A5. Additional Ablations and Detailed Tables

RADSeg modules generalize across RADIO. Tab. A.4 demonstrates that the proposed SCRA and SCGA modules generalize across all RADIO versions and model sizes, with consistent improvements ranging from +3.4 to +6.9 avg mIoU. Gains are largest on Cityscapes (up to +11.2), whose high-resolution images require more sliding windows and thus benefit most from SCGA's cross-window consistency. V2.5 variants also see larger gains overall, where weaker base spatial locality leaves more room for refinement.

Scaling the attention correlation matrix has a notable impact. We ablate temperature parameters τ_{scra} and τ_{scga} for SCRA and SCGA at mid resolution across 2D datasets. Fig. A.1 highlights the importance of scaling the attention correlation matrix to sharpen the attention to seman-

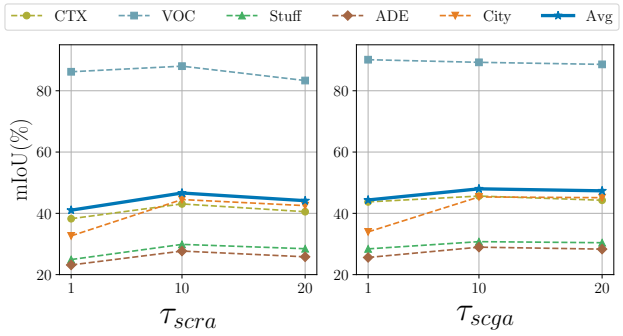

Figure A.1. Ablation study on different temperature parameters τ_{scra} and τ_{scga}. Left plot shows performance as we change τ_{scra} without SCGA. Right plot uses $\tau_{scra} = 10$ and varies τ_{scga}. Overall $\tau_{scra} = \tau_{scga} = 10$ yield the best results on average.

tically similar patches and similarly for global aggregation. $\tau_{scra} = \tau_{scga} = 10$ yield the best results and is what we use for all experiments.

Table A.5. Increasing SCRA recursion does not help (w/SCGA).

Iterations	1	2	3
Avg mIoU	**48.0**	47.5	47.0

Increasing recursion depth of SCRA does not help. As shown in Tab. A.5, increasing recursion beyond a single iteration yields diminishing returns, with avg mIoU dropping from 48.0 to 47.0 at 3 iterations, likely due to over-reliance on the correlation signal.

Table A.6. RADSeg supports single-pass high-res inference thanks to RADIO's flexible resolution support, unlike CLIP-based methods that require sliding windows. (mIoU shown)

Method	CTX	VOC	Stuff	ADE	City	Avg
Sliding Window	45.64	**89.28**	30.76	28.96	**45.35**	**48.00**
Single Inference	**47.52**	89.02	**31.47**	**29.51**	41.96	47.90
Single Infer w/o SCGA	**47.52**	87.81	31.29	28.81	43.83	47.85

RADSeg supports single-pass high-resolution inference. As shown in Tab. A.6, single-pass inference only entails a -0.1 avg mIoU drop. Unlike CLIP-based baselines, whose fixed positional embeddings necessitate sliding windows at higher resolutions, RADIO's cropped position embeddings (CPE) natively accept arbitrary resolutions, making single-pass inference practical. Notably, single inference outperforms sliding windows on most datasets, with only Cityscapes—the highest resolution dataset requiring the most windows—benefiting from the sliding window approach. **SCGA remains beneficial even without windows by refining global feature consistency**.

Finally, for reference, we provide per-dataset details for Tab. 3 in Tab. A.7 and per-dataset details for Tab. 5 in Tab. A.8.

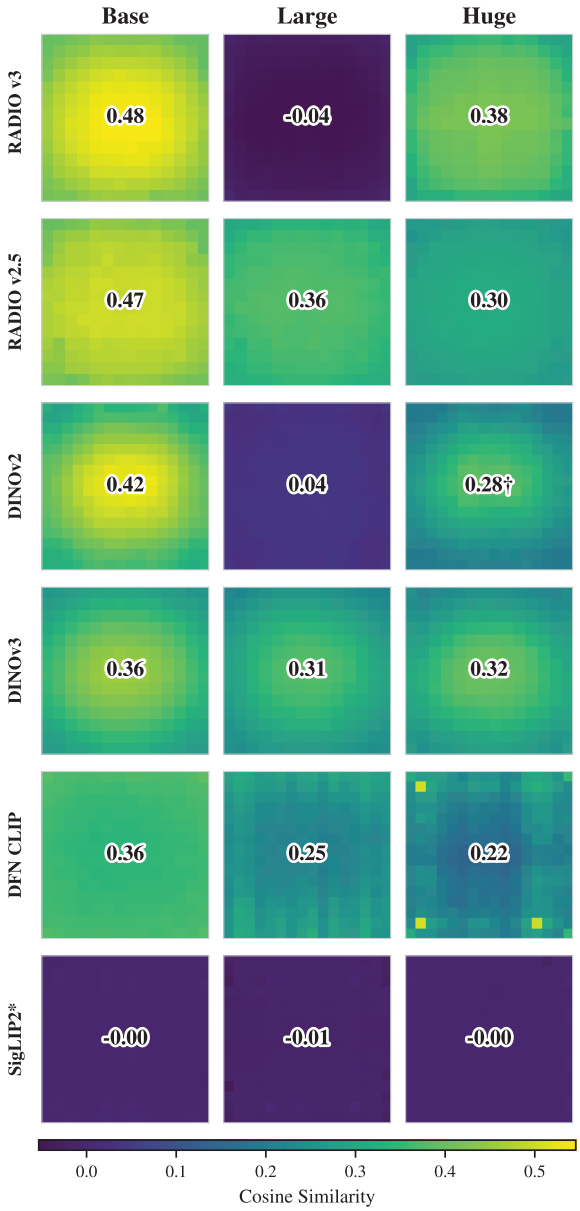

Figure A.2. Average CLS-to-patch cosine similarity across six vision model families at three scales. Lower-capacity models consistently exhibit stronger CLS-to-patch alignment. RADIOv3-L and DINOv2-L are notable outliers with near-zero alignment. SigLIP2 shows no alignment due to its MAP pooling architecture. †Giant variant (1.1B params). *Pooled output vs. patch features.

A6. CLS-Patch Alignment Analysis

A key property exploited by RADSeg is the alignment between the CLS summary token and output patch features in

Table A.7. Expanded version of Tab. 3. Ablation of different RADIO language adapters across model sizes. Mid resolution is used. **RADIOv3-base with the SigLIP2 CLS adaptor has the best performance across datasets and model sizes**.

Methods	Base						Large						Huge					
	CTX	VOC	Stuff	ADE	City	Avg	CTX	VOC	Stuff	ADE	City	Avg	CTX	VOC	Stuff	ADE	City	Avg
Rv2.5-SigLIP$_{cls}$	34.24	84.99	24.66	23.11	32.78	39.96	31.70	83.88	23.68	21.00	32.49	38.55	31.12	79.04	23.00	21.92	29.78	36.97
Rv2.5-CLIP$_{cls}$	33.42	86.55	23.56	22.37	31.24	39.43	31.09	84.57	22.68	20.10	30.53	37.79	27.81	79.70	21.08	20.03	25.65	34.85
Rv2.5-SigLIP$_{patch}$	0.31	1.00	0.06	0.14	0.09	0.32	0.27	0.97	0.06	0.12	0.14	0.31	0.37	1.06	0.06	0.17	0.13	0.36
Rv3-SigLIP$_{cls}$	**40.40**	**88.07**	**27.41**	**27.30**	**39.94**	**44.62**	**1.00**	**1.46**	**0.52**	**0.15**	**1.18**	**0.86**	**35.23**	**88.35**	**24.55**	**24.34**	**33.74**	**41.24**
Rv3-CLIP$_{cls}$	37.90	86.89	26.36	24.48	35.99	42.32	0.67	1.71	0.07	0.12	1.17	0.75	29.48	84.07	21.85	21.39	27.16	36.79
Rv3-SigLIP$_{patch}$	0.11	2.47	0.05	0.14	0.89	0.73	0.12	2.50	0.04	0.14	0.98	0.76	0.12	0.84	0.05	0.12	0.14	0.25

Table A.8. Expanded version of Tab. 5. Ablation study showing RADIOv3's ability to empower existing approaches. Mid resolution is used. ProxyRADIO-D/S refer to using DINOv2 and SAM adapted feature maps respectively for the proxy attention. **RADIO can improve mIoU for different CLIP-based baselines**.

Methods	CTX	VOC	Stuff	ADE	City	Avg
NACLIP [13]	34.70	80.28	23.65	17.78	35.98	38.48
NARADIO	40.85	86.25	27.93	27.19	40.47	44.54
ProxyCLIP [20]	37.76	80.24	26.4	19.65	40.28	40.87
ProxyRADIO-D	41.62	86.52	29.28	26.90	42.87	45.44
ProxyRADIO-S	41.99	87.90	29.33	27.00	44.04	46.05
ResCLIP [44]	36.52	85.79	24.91	18.43	36.02	40.33
ResRADIO	41.79	87.96	28.74	26.63	40.77	45.18
RADSeg	**45.64**	**89.28**	**30.76**	**28.96**	**45.35**	**48.00**

the RADIO backbone. When this alignment is strong, projecting patch features through the language adaptor head—which is trained to map the CLS token to the text embedding space—yields spatially dense, language-aligned features suitable for zero-shot segmentation. When this alignment is weak, the adaptor projection produces representations that do not correspond to meaningful text embeddings, and segmentation performance degrades.

To quantify this, we compute the average cosine similarity between each output patch feature and the CLS token across 1,000 ImageNet validation images for six model families at three scales (Fig. A.2): RADIOv3 and RADIOv2.5 (using their respective language adaptors), DINOv2 with registers, DINOv3, DFN CLIP, and SigLIP2.

Two findings emerge. First, RADIOv3-L exhibits near-zero CLS-patch similarity (-0.04), indicating a lack of alignment between the CLS summary and spatial patch representations. This directly explains the zero segmentation performance observed for this variant in Tab. 3. A similar phenomenon appears in DINOv2-L (0.04); understanding why the Large scale specifically loses alignment in both families may require reproducing/retraining these models and hence is beyond our computational resources and scope.

Second, across model families, lower-capacity backbones consistently exhibit stronger CLS-patch alignment (e.g., C-RADIO v3: 0.48 Base vs. 0.38 Huge; RADIO v2.5: 0.47 vs. 0.30; DFN CLIP: 0.36 vs. 0.22). We hypothesize that models with limited capacity are constrained to share representational structure between the CLS and patch tokens, naturally producing stronger alignment. This is consistent with observations in single-encoder vision models [4] and explains the counterintuitive, yet useful, result that base-scale RADIO backbones can outperform larger variants for zero-shot segmentation.

Note that the spatial pattern of alignment also exhibits a consistent center bias across models, reflecting the tendency of the CLS token to summarize the central, salient entity in the image. Moreover, SigLIP2 shows near-zero similarity (~ 0.00) across all scales, as its MAP pooling head applies learned cross-attention projections that place the pooled output in a different subspace from the output patch features.

A7. Additional Qualitative Results

We provide additional 2D and 3D qualitative comparisons in Fig. A.4 and Fig. A.5, further illustrating the strengths of **RADSeg** over existing open-vocabulary segmentation methods. In 2D, **RADSeg** yields cleaner boundaries and more accurate segmentations in both cluttered indoor scenes and complex urban layouts. In 3D, it shows strong multi-view consistency, producing coherent semantic voxels with far fewer outliers and mislabeled regions than the baselines. These visualizations reinforce the ability of **RADSeg**, with its proposed components, to suppress noise and enhance object localizations across various scenarios.

RADSeg modules qualitatively improve feature map and segmentation quality. To demonstrate the effectiveness of the components of **RADSeg**, we augment Tab. 4 with a qualitative visualization. Fig. A.3 qualitatively illustrates the effectiveness of SCRA and SCGA in suppressing noise in the feature map and segmentation as well as reducing windowing artifacts. The effect is particularly pronounced in higher resolution datasets like Cityscapes. While RADIO-SAM refinement is a post-segmentation process that cannot refine features, it is able to provide higher fidelity masks in many cases. Overall, the proposed **RADSeg** and **RADSeg+** components demonstrate quantitative and qualitative improvements.

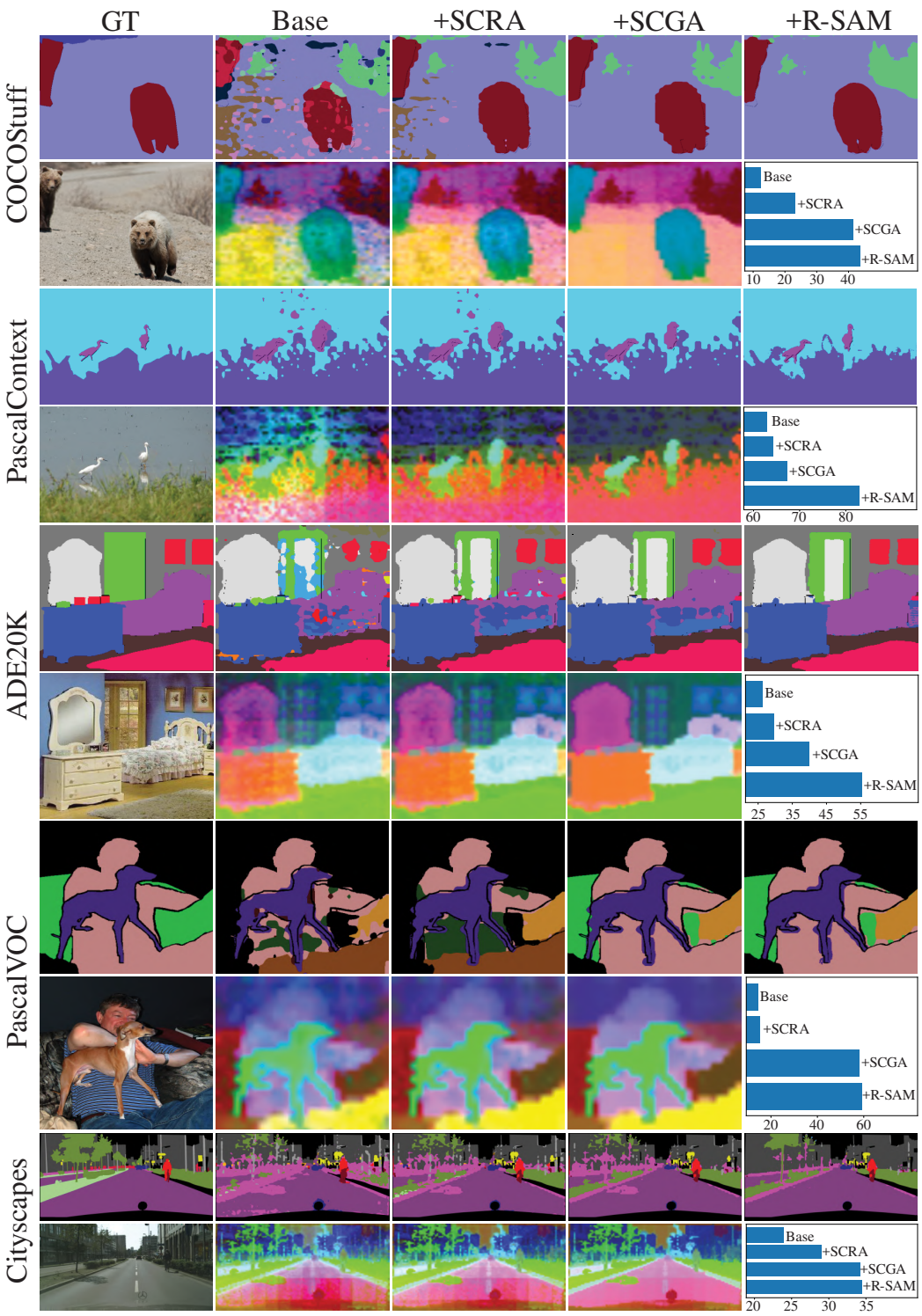

Figure A.3. Qualitative comparison of the contribution of each **RADSeg** component to feature and prediction quality. For each 2D dataset we show two rows: the first demonstrates how adding **RADSeg** components progressively improves segmentation output, while the second (Showing first 3 PCA components) illustrates how SCRA and SCGA enhance feature map quality and mitigate windowing artifacts. The accompanying bar plot links these visual trends to per-sample mIoU scores. Overall, the proposed **RADSeg** components yield clear improvements in both segmentation accuracy and feature-map fidelity.

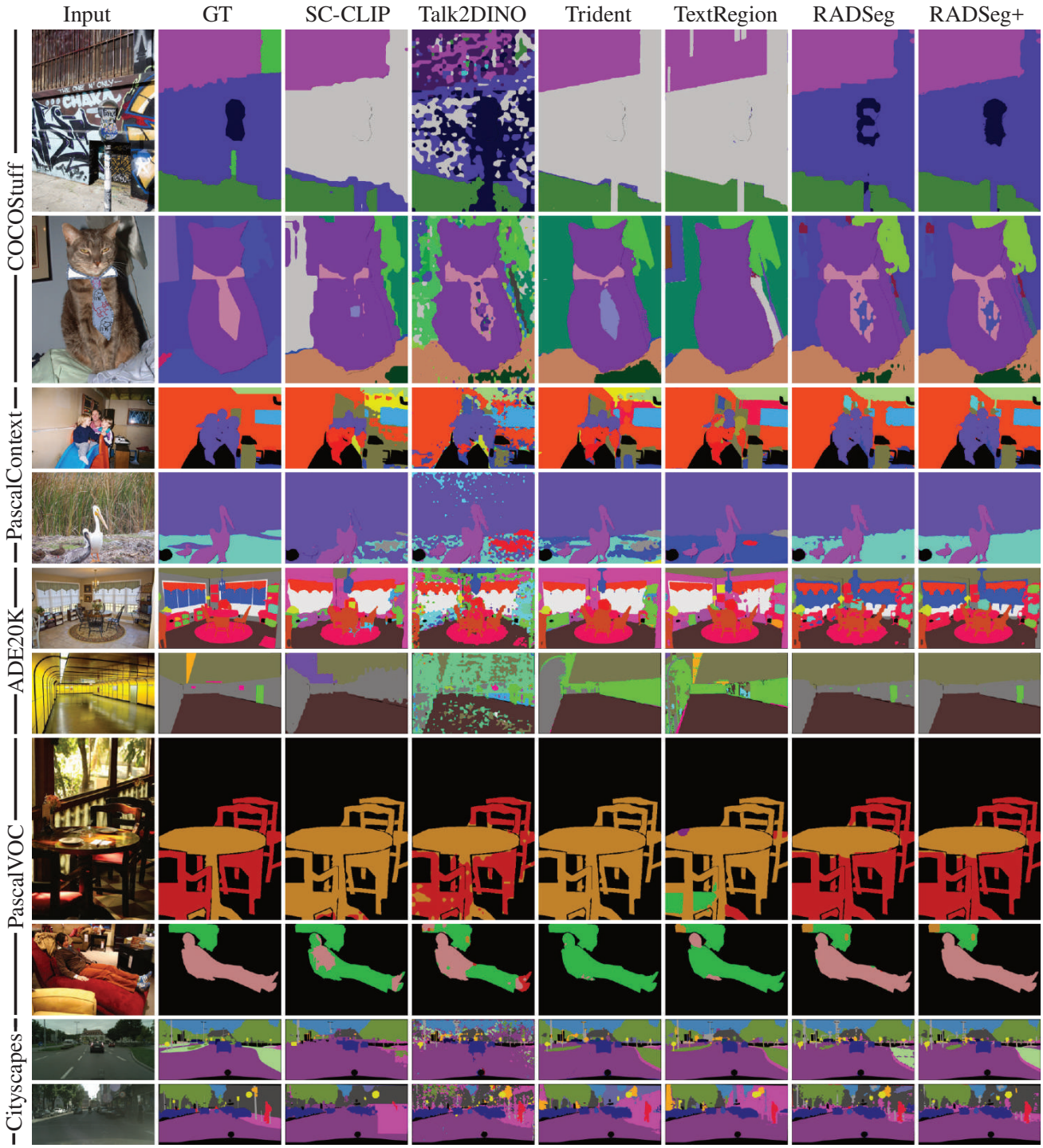

Figure A.4. Additional visualizations of 2D semantic segmentation results generated by **RADSeg**, **RADSeg+** and the most competitive baselines on images from all the benchmarks. Along with achieving SOTA mIoU for 2D OVSS, **RADSeg** and **RADSeg+** produce more precise segmentation maps and sharper object boundaries for both single-object as well as multi-object complex scenes.

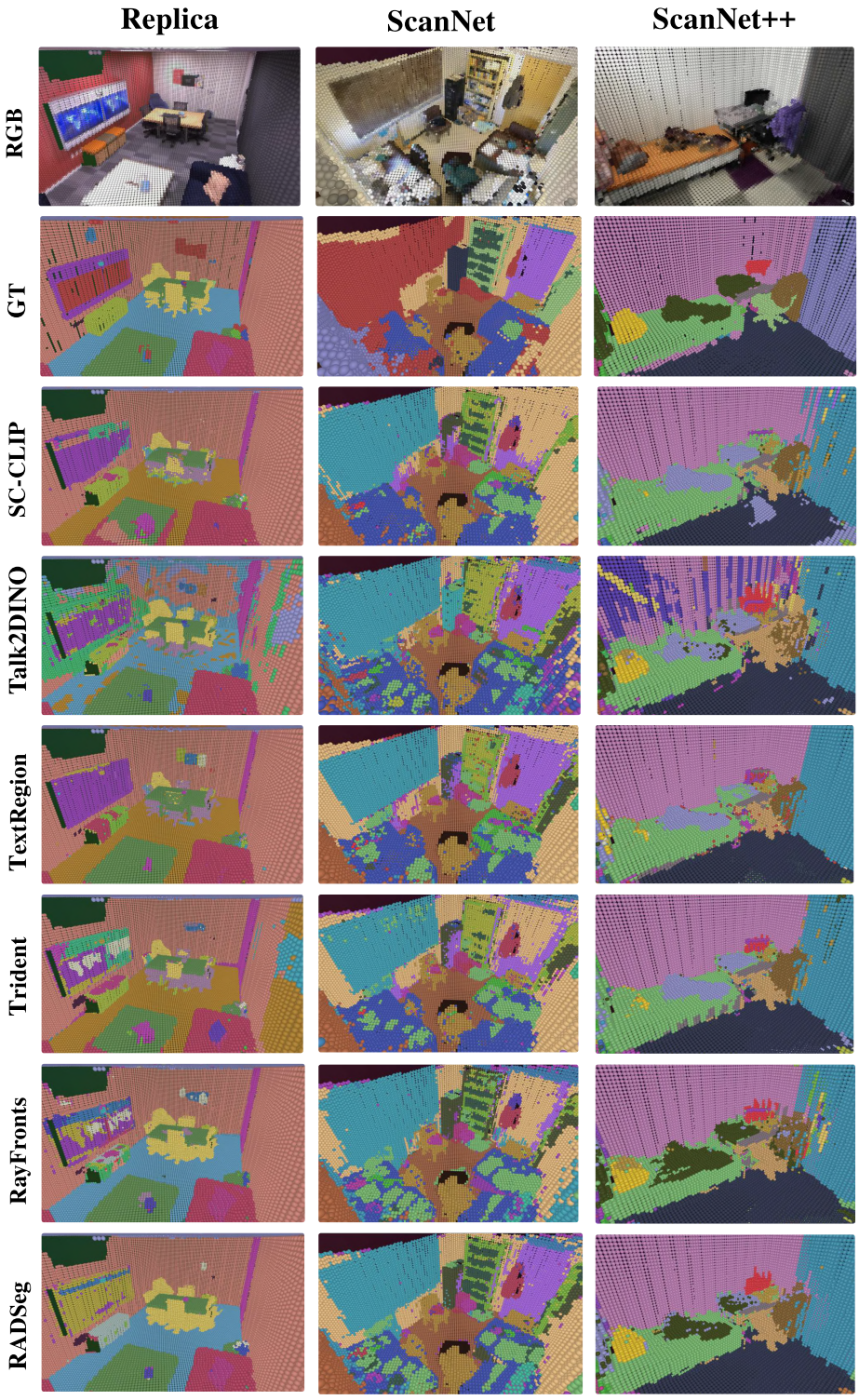

Figure A.5. Sample visualizations of 3D semantic segmentation results generated by **RADSeg** and all the baselines for scenes from Replica (scene-`office2`), ScanNet (scene-`0378`), and ScanNet++ (scene-`ea42cd27e6`. "RGB" and "GT" refer to the RGB scene reconstruction and Ground Truth semantics for each corresponding scene. **RADSeg** achieves SOTA mIoU for 3D OVSS and produces cleaner and more accurate semantic voxels.