Supplemental Materials for CoinSeg: Contrast Inter- and Intra- Class Representations for Incremental Segmentation

Contents

1. More Experimental Details

1.1. Overlapped & disjoint setup

We provide a detailed explanation of the *overlapped* and *disjoint* settings in this section. The overlapped setting allows pixels in the samples from the sub-dataset $\mathcal{D}^t$ to belong to any classes, including past classes from learning step 1 to $t-1$ (*i.e.*, $\mathcal{C}^{1:t-1}$), current classes ($\mathcal{C}^t$), and future classes. However, only the classes in $\mathcal{C}^t$ are annotated in $\boldsymbol{y}^t$. Moreover, images that contain multiple classes may appear in several learning steps with varying annotations. In contrast, the disjoint setting studied in prior works, such as [1, 9, 10], comprises a non-overlapping subset of datasets. Each learning step contains a unique $\mathcal{C}^t$, and its pixels only belong to classes seen in $\mathcal{C}^{1:t-1}$ or $\mathcal{C}^t$. Notably, the overlapped setup is more realistic as it imposes a weaker restriction on the data than the disjoint setup.

1.2. Reproducibility

In all experiments, CoinSeg adopts the same mask proposals and the number of proposals $N = 100$, according to prior practice as in the class incremental semantic segmentation (CISS) method MicroSeg [11]. We adopt the official `Pre-trained Model` of Swin Transformer [7] for all experimental results. In the case of CoinSeg-M, which is CoinSeg equipped with a memory sampling strategy, we implemented the strategy using the official code of the prior CISS work SSUL [2]. To be more specific, the strategy is based on random sampling from the training dataset, while also ensuring that at least one sample of every seen class is present in the memory bank.

2. Additional Experimental Analysis

2.1. Detailed experimental results of CoinSeg

Tab. S1 shows the detailed experimental results over each class in the Pascal VOC 2012 [4] dataset.

2.2. CoinSeg with ResNet101 backbone

To further validate the effectiveness of our proposed approach and ensure a fair comparison with other methods from a different perspective, we conducted additional experiments with the CoinSeg method using ResNet101 as the backbone. The results are presented in Tab. S2, From which we can see, even with a relatively weaker backbone, the results indicate that our approaches, including CoinSeg and CoinSeg-M, are still competitive, and achieve state-of-the-art performance across multiple incremental scenarios.

2.3. Experimental results of disjoint setting

To provide a thorough comparison with prior works on CISS, we further present the experimental results of our proposed methods, CoinSeg and CoinSeg-M, in the disjoint setting in Tab. S3. We evaluate our methods using both the ResNet101 and Swin-B backbones. The results show that our methods achieve state-of-the-art performance, surpassing prior works on CISS.

2.4. Ablation study of contrast intra-class diversity

As highlighted in the main paper, strategies aimed at contrasting intra-class diversity are critical for learning robust

Table S1. Detailed experimental results of CoinSeg over each class.

VOC 10-1	bg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
	88.0	89.9	40.8	94.1	76.3	88.5	91.1	89.7	96.2	48.4	77.2
	table	dog	horse	mbike	person	plant	sheep	sofa	train	TV	**mIoU**
	37.8	89.4	75.3	83.7	85.5	47.1	59.5	27.4	74.8	53.2	**72.5**
VOC 15-1	bg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
	90.3	89.8	43.5	95.2	79.3	88.0	91.2	90.8	96.3	47.1	78.1
	table	dog	horse	mbike	person	plant	sheep	sofa	train	TV	**mIoU**
	68.5	93.3	91.0	92.0	89.7	43.7	64.1	27.0	72.7	55.4	**75.5**
VOC 19-1	bg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
	93.9	91.8	43.0	94.5	76.4	87.2	95.0	89.6	96.5	49.1	91.5
	table	dog	horse	mbike	person	plant	sheep	sofa	train	TV	**mIoU**
	65.9	89.7	90.1	90.3	90.9	63.0	93.0	54.0	86.0	44.8	**79.8**
VOC 15-5	bg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
	91.3	90.9	43.8	95.4	74.5	85.2	90.0	91.3	95.2	49.1	79.5
	table	dog	horse	mbike	person	plant	sheep	sofa	train	TV	**mIoU**
	66.7	90.5	91.0	90.9	90.1	55.2	72.4	31.3	84.0	73.1	**77.6**
VOC 2-2	bg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
	89.1	85.4	35.8	53.2	56.9	76.7	88.5	80.3	88.0	18.3	64.4
	table	dog	horse	mbike	person	plant	sheep	sofa	train	TV	**mIoU**
	43.0	59.2	64.9	79.8	80.8	60.2	51.8	32.4	74.1	67.9	**64.3**

Table S2. Comparison with state-of-the-art methods on Pascal VOC 2012 with ResNet101. Joint is the upperbound.

Method	Backbone	**VOC 10-1 (11 steps)**			**VOC 15-1 (6 steps)**			**VOC 19-1 (2 steps)**			**VOC 15-5 (2 steps)**			**VOC 2-2 (10 steps)**		
		0-10	11-20	all	0-15	16-20	all	0-19	20	all	0-15	16-20	all	0-2	3-20	all
Joint	ResNet101	82.1	79.6	80.9	82.7	75.0	80.9	81.0	79.1	80.9	82.7	75.0	80.9	76.5	81.6	80.9
LwF-MC [6]	ResNet101	4.7	5.9	4.9	6.4	8.4	6.9	64.4	13.3	61.9	58.1	35.0	52.3	3.5	4.7	4.5
ILT [8]	ResNet101	7.2	3.7	5.5	8.8	8.0	8.6	67.8	10.9	65.1	67.1	39.2	60.5	5.8	5.0	5.1
MiB [1]	ResNet101	12.3	13.1	12.7	34.2	13.5	29.3	71.4	23.6	69.2	76.4	50.0	70.1	41.1	23.4	25.9
SDR [9]	ResNet101	32.1	17.0	24.9	44.7	21.8	39.2	69.1	32.6	67.4	57.4	52.6	69.9	13.0	5.1	6.2
PLOP [3]	ResNet101	44.0	15.5	30.5	65.1	21.1	54.6	75.4	37.4	73.5	75.7	51.7	70.1	24.1	11.9	13.7
RCIL [10]	ResNet101	55.4	15.1	34.3	70.6	23.7	59.4	68.5	12.1	65.8	78.8	52.0	72.4	28.3	19.0	19.4
SSUL [2]	ResNet101	71.3	46.0	59.3	77.3	36.6	67.6	77.7	29.7	75.4	77.8	50.1	71.2	62.4	42.5	45.3
SSUL-M[2]	ResNet101	74.0	53.2	64.1	78.4	49.0	71.4	77.8	49.8	76.5	78.4	55.8	73.0	58.8	45.8	47.6
MicroSeg [11]	ResNet101	72.6	48.7	61.2	80.1	36.8	69.8	78.8	14.0	75.7	80.4	52.8	73.8	61.4	40.6	43.5
MicroSeg-M [11]	ResNet101	**77.2**	57.2	**67.7**	81.3	52.5	74.4	79.3	62.9	78.5	**82.0**	59.2	**76.6**	60.0	50.9	52.2
CoinSeg (Ours)	ResNet101	73.7	45.0	60.1	80.6	36.2	70.1	**80.7**	29.4	78.3	80.8	31.0	68.9	68.3	46.2	49.4
CoinSeg-M (Ours)	ResNet101	71.1	**60.2**	65.9	**81.5**	**54.0**	**75.0**	80.3	**44.8**	**78.6**	81.3	**61.3**	**76.6**	**66.7**	**53.9**	**55.7**

representations. In this regard, we provide quantitative evidence to support this claim. Specifically, Tab. S5 reveals that using $\mathcal{L}_{int}$ only yields a marginal improvement in performance. This suggests that while assigning a representation to each class via average pooling may be effective to some extent, it is not the optimal strategy, and the improvement in performance is limited. On the other hand, combining $\mathcal{L}_{int}$ with $\mathcal{L}_{itr}$ leads to a significant improvement in overall performance. These findings indicate that contrastive learning of intra-class diversity is an effective approach for robust representation learning.

2.5. Ablation to pseudo-labeling strategy in $\mathcal{L}_{int}$

As detailed in the main paper Sec. 3.2, we employ pseudo-labeling within $\mathcal{L}_{int}$ to extract categories learned in previous learning steps. To illustrate its impact, we report the performance of $\mathcal{L}_{int}$ just with the current class labels in Tab. S4 Right ('GT' row). We can see that the proposed pseudo-labeling (PL) boosts the performance.

2.6. Discussions of adaption of Swin Transformer

Our proposed contrast intra- and inter-class diversity representation approach places a greater emphasis on local information. In contrast to conventional CNN-based architectures, the Swin Transformer offers superior feature representation

Table S3. Experimental results on Pascal VOC 2012 for *disjoint* setup. †: CoinSeg with Swin-B backbone.

Method	VOC 15-1 (6 steps)			VOC 19-1 (2 steps)			VOC 15-5 (2 steps)		
	0-15	16-20	all	0-19	20	all	0-15	16-20	all
LwF-MC [6]	4.5	7.0	5.2	63.0	13.2	60.5	67.2	41.2	60.7
ILT [8]	3.7	5.7	4.2	69.1	16.4	66.4	63.2	39.5	57.3
MiB [1]	46.2	12.9	37.9	69.6	25.6	67.4	71.8	43.3	64.7
SDR [9]	59.4	14.3	48.7	70.8	31.4	68.9	74.6	44.1	67.3
PLOP [3]	57.9	13.7	46.5	75.4	38.9	73.6	71.0	42.8	64.3
RCIL [9]	66.1	18.2	54.7	68.9	15.0	66.3	75.0	42.8	67.3
SSUL [2]	74.0	32.2	64.0	77.4	22.4	74.8	76.4	45.6	69.1
SSUL-M [2]	76.5	43.4	68.6	77.6	43.9	76.0	76.5	48.6	69.8
MicroSeg [11]	73.7	24.1	61.9	80.6	16.0	77.4	77.4	43.4	69.3
MicroSeg-M [11]	80.0	47.6	72.3	81.1	45.1	79.4	80.7	55.2	74.7
CoinSeg (Ours)	75.6	30.9	64.9	80.5	25.1	77.9	79.6	43.8	71.1
CoinSeg-M (Ours)	80.9	47.4	72.9	80.8	50.6	79.4	81.7	58.9	76.3
CoinSeg† (Ours)	**82.0**	46.1	73.4	82.0	34.0	80.2	82.1	55.3	75.7
CoinSeg-M† (Ours)	**82.0**	**49.6**	**74.3**	**82.6**	**66.3**	**81.8**	**82.9**	**61.7**	**77.9**

Table S4. Ablations to pseudo-labeling in $\mathcal{L}_{int}$. GT: ground truth, PL: pseudo label.

label	VOC 15-1 (6 steps)		
	0-15	16-20	all
GT	81.6	54.3	75.1
PL	82.7	52.5	**75.5**

Table S5. Ablation study of Contrast inter- & intra- class representations, λ_r. All experiments are conducted on VOC 15-1.

Method	$\mathcal{L}_{int}$	$\mathcal{L}_{itr}$	VOC 15-1 (6 steps)		
			0-15	16-20	all
CoinSeg	✗	✗	80.4	43.7	71.6
	✓	✗	80.8	45.9	72.4 (+0.8)
	✓	✓	82.7	52.5	**75.5** (+3.9)

capabilities for local patches [5, 7]. Hence, we choose the Swin Transformer as the backbone of the model. Tab. S6 show the comparison of both backbones, ResNet101 and Swin Transformer. The results show that CoinSeg achieves a significant improvement in performance with the Swin Transformer, particularly in more challenging long-term scenarios such as VOC 10-1 and 2-2.

Meanwhile, due to the freeze strategy, prior methods may not be able to fully take advantage of the backbone's performance. As a result, the performance improvement of prior methods is limited when replacing ResNet101 with Swin Transformer as the backbone. This conclusion can be drawn by comparing the experimental results presented in the main paper. However, our proposed CoinSeg method shows significant performance improvement when using Swin Transformer as the backbone.

It is important to note that while applying the Swin Transformer results in improved performance, our approach does not depend solely on a stronger backbone. As shown in Tab. S2, CoinSeg achieves state-of-the-art performance even when using the ResNet101 backbone.

2.7. Search of hyper-parameters

Here we present the results of the hyper-parameters search, including λ_r, λ_c, λ_{lr} and τ. We have done the parameter search on five orders of magnitude on VOC 15-1, for λ_r λ_c and λ_{lr}. The results of Tab. S7 show λ_r, the hyper-parameter to balance regularization constraints, is not sensitive to the choice of these parameters. Even if the λ_r varies by several orders of magnitude (10^{-3} to 1), the performance is stable. The results show that our proposed method, CoinSeg, is effective over a wide range of choices of hyper-parameters. Tab. S8 presents the parameter search of λ_c, the hyper-parameter weight of contrast inter- and intra-class representations (*Coin*). And Tab. S9 is the performance comparison with different choices of λ_{lr} in flexible initial learning rate. Additionally, we also report the more detailed experimental results in different incremental scenario (VOC 10-1) in Tab. S10. The results lead to similar conclusions. Furthermore, we have done a parameter search for threshold τ for pseudo-label in Tab. S11. The conclusions are still similar.

Additionally, we also conduct a hyperparametric search within a similar magnitude, as most previous works did, as shown in Tab. S12. Our method shows a performance fluctuation of only 0.4% in this perspective. This demonstrates the robustness of our proposed CoinSeg with respect to hyperparameters. As a summary, we choose $\lambda_r = 0.1$, $\lambda_c = 0.01$, $\lambda_{lr} = 10^{-3}$ and $\tau = 0.7$ for the best performance.

Table S6. Comparisons of CoinSeg with different backbones. †: Re-implemented with Swin-B backbone.

Method	Backbone	**VOC 10-1 (11 steps)**			**VOC 15-1 (6 steps)**			**VOC 19-1 (2 steps)**			**VOC 15-5 (2 steps)**			**VOC 2-2 (10 steps)**		
		0-10	11-20	all	0-15	16-20	all	0-19	20	all	0-15	16-20	all	0-2	3-20	all
CoinSeg	Resnet101	73.7	45.0	60.1	80.6	36.2	70.1	80.7	29.4	78.3	80.8	31.0	68.9	68.3	46.2	49.4
CoinSeg†	Swin-B	80.0	63.4	72.5	82.7	52.5	75.5	81.5	44.8	79.8	82.1	63.2	77.6	70.1	63.3	64.3

Table S7. Parameter search of hyper-parameter of regularization constraints, λ_r. All experiments are conducted on VOC 15-1.

Method	λ_r	**VOC 15-1 (6 steps)**		
		0-15	16-20	all
CoinSeg	10^{-3}	80.9	53.3	74.3
	0.01	81.9	52.6	74.9
	0.1	82.7	52.5	**75.5**
	1	81.7	50.6	74.3
	10	80.8	50.4	73.5

Table S8. Parameter search of hyper-parameter to balance *Coin*, λ_c. All experiments are conducted on VOC 15-1.

Method	λ_c	**VOC 15-1 (6 steps)**		
		0-15	16-20	all
CoinSeg	10^{-3}	81.4	48.9	73.7
	0.01	82.7	52.5	**75.5**
	0.1	82.4	51.8	75.1
	1	81.8	49.9	74.2
	10	80.6	46.6	72.5

2.8. Reproduction of past methods

In this paper, a portion of the experimental results are obtained through reproducing previous methods, e.g., the performance of previous methods equipped with the Swin-B backbone. Note that we have made extensive efforts to optimize the performance of previous methods on the Swin transformer, to facilitate a fair comparison. Tab. S13 show the parameter tuning the most crucial hyperparameter, K, in previous method MicroSeg.

3. More Qualitative Results

In addition to the qualitative results shown in the main paper, here in Fig. 1 shows more qualitative results of ADE20K [12] dataset. We conducted a long-term incremental scenario, ADE 100-10, consisting of six steps to better evaluate the effectiveness of CoinSeg. To provide a more comprehensive analysis, we present qualitative results of multiple scenes, including indoor and outdoor, things and stuff. The results show that CoinSeg rarely forgets old concepts during incremental learning steps, validating its effectiveness in long-term incremental scenarios.

In addition, we provide more qualitative results in Fig. 2 and Fig. 3, for VOC 15-1. In some samples, novel classes appear in the incremental steps, and the results demonstrate that our proposed CoinSeg can effectively adapt to these new categories. Meanwhile, in samples which only contians base classes, the predictions remain stable through all learning steps, which indicates CoinSeg alleviating forgetting.

Table S9. Parameter search of hyper-parameter in flexible initial learning rate, λ_{lr}. All experiments are conducted on VOC 15-1.

Method	λ_{lr}	**VOC 15-1 (6 steps)**		
		0-15	16-20	all
CoinSeg	10^{-4}	83.0	48.4	74.5
	10^{-3}	82.7	52.5	**75.5**
	0.01	82.6	51.1	75.1
	0.1	82.3	50.7	74.8
	1	81.3	49.6	73.8

Table S10. More detailed hyperparameter search of λ_{lr} on VOC 10-1, within similar magnitudes.

Method	λ_{lr}	**VOC 10-1 (11 steps)**		
		0-10	11-20	all
CoinSeg	1×10^{-4}	80.4	58.6	69.9
	5×10^{-4}	80.2	59.1	70.2
	1×10^{-3}	80.1	60.0	**70.5**
	2×10^{-3}	79.8	59.5	70.1
	5×10^{-3}	79.0	59.2	69.6

Table S11. Parameter search of hyper-parameter: threshold τ for pseudo-label. All experiments are conducted on VOC 15-1.

Method	τ	**VOC 15-1 (6 steps)**		
		0-15	16-20	all
CoinSeg	0.1	81.4	47.5	73.3
	0.3	81.3	49.7	73.8
	0.5	82.0	50.6	74.5
	0.7	82.7	52.5	**75.5**
	0.9	81.1	51.9	74.1

Table S12. Hyperparameter search of λ_c within a similar magnitude, in VOC 15-1.

λ_c	VOC 15-1 (6 steps)		
	0-15	16-20	all
0.75×10^{-2}	82.5	53.0	75.5(+0.0)
1×10^{-2}	82.7	52.5	**75.5**
3×10^{-2}	82.7	52.3	75.4(-0.1)
5×10^{-2}	82.6	51.4	75.1(-0.4)
7×10^{-2}	82.6	52.0	75.2(-0.3)

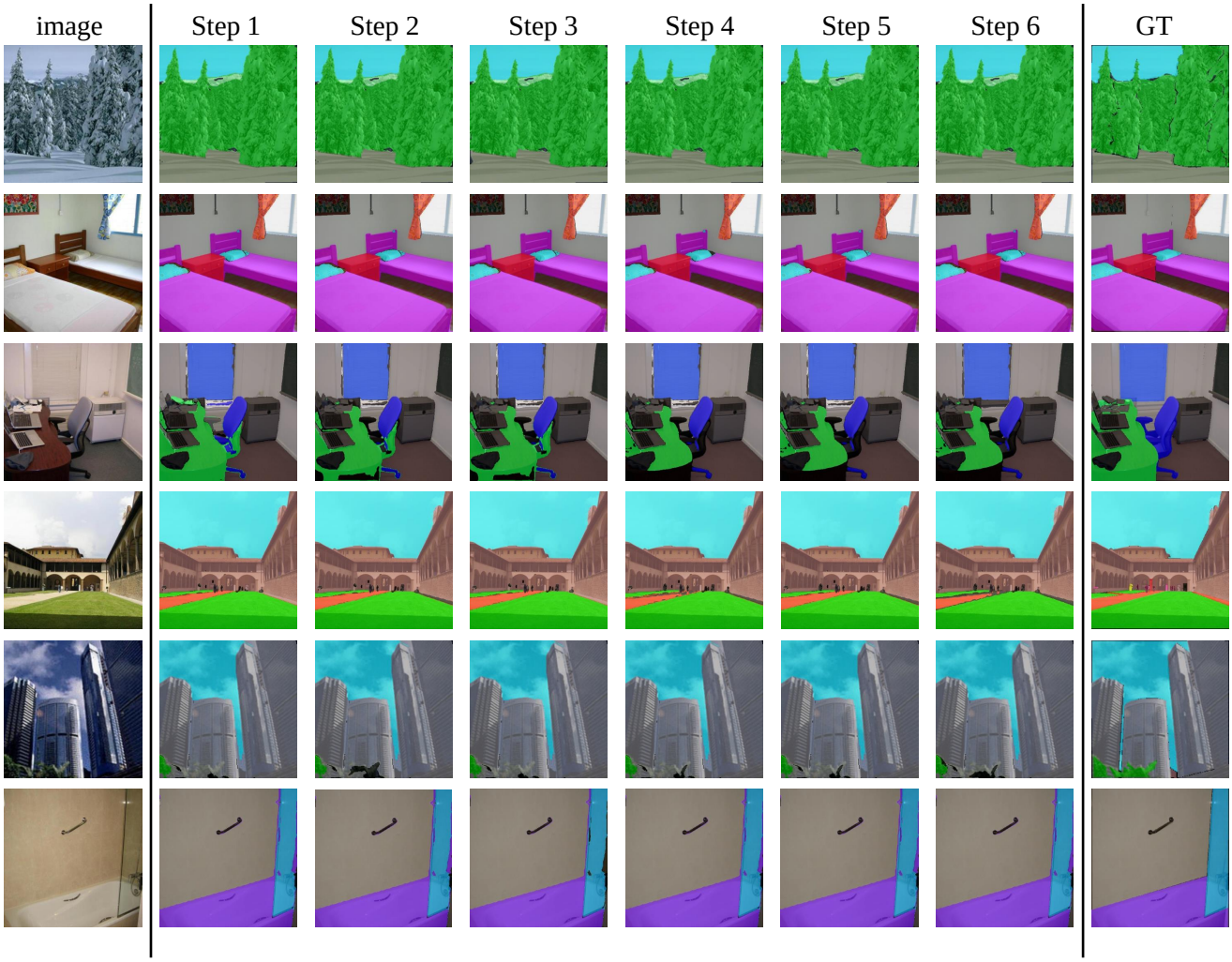

Figure 1. Qualitative analysis for ADE 100-10.

Table S13. Hyperparameter search of K in MicroSeg with Swin-B backbone (Left), and λ_c within a similar magnitude (Right).

K	VOC 15-1 (6 steps)		
	0-15	16-20	all
1	78.8	37.1	68.9
3	79.1	38.6	69.5
5	80.5	40.8	**71.0**
7	81.4	35.9	70.6
9	81.2	36.1	70.4

Limitations, future work and social impact Although our approach, CoinSeg, achieves state-of-the-art performance in numerous benchmarks, forgetting still exists in long-term scenarios. For future work, it might be interesting to explore how a well-designed deep learning model architecture can be better applied to tackle the long-term incremental learning tasks.

All of our experimental results are produced on public datasets and the research of CoinSeg has no obvious AI ethical issues, to the best of our knowledge. But training deep learning models does have some potential environmental impact due to the power consumption. We hope that our proposed approach can help researchers to conduct further exploration of class incremental semantic segmentation.

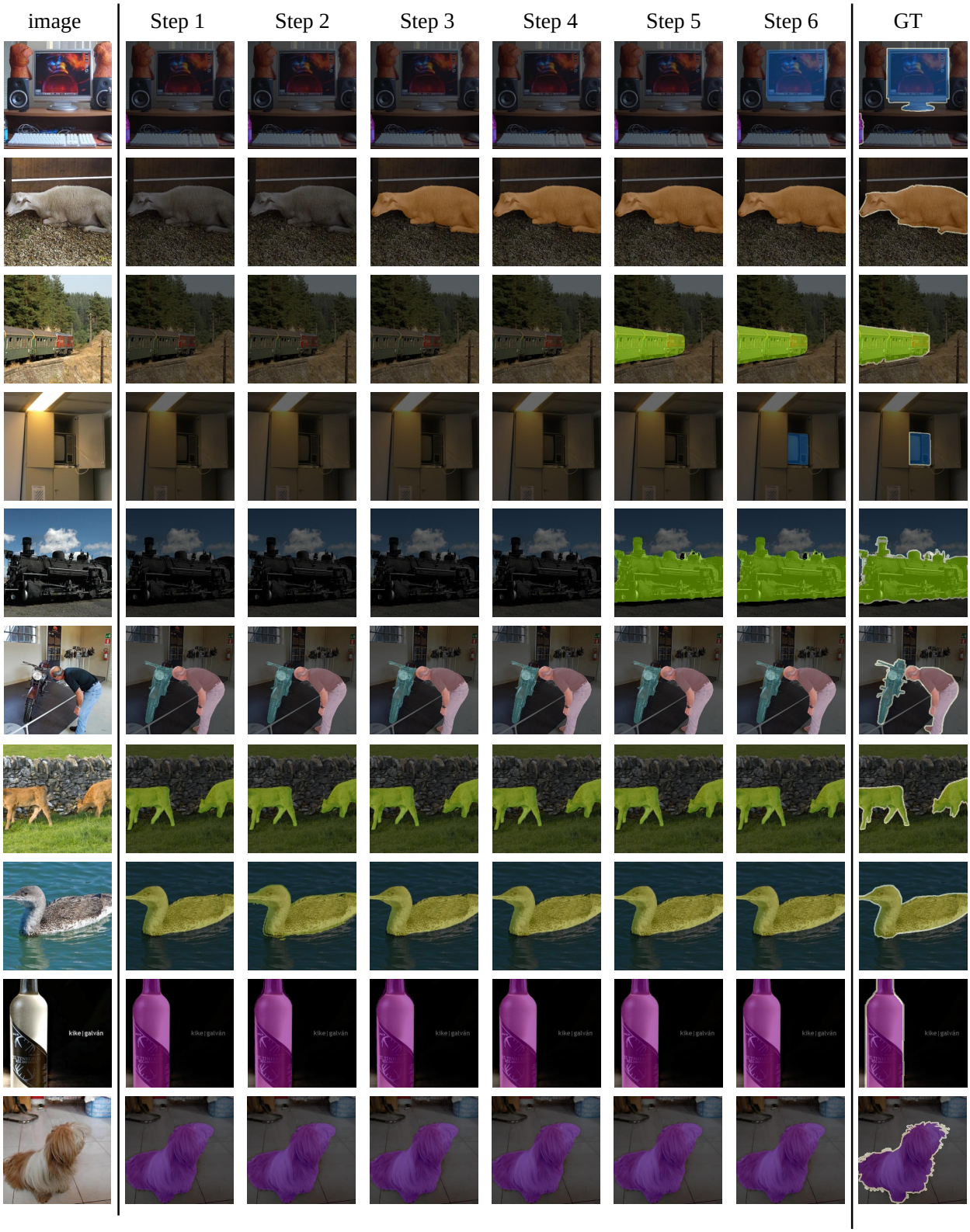

Figure 2. More qualitative analysis of CoinSeg. The white borderline is **ignore label** in Pascal VOC 2012 dataset.

Figure 3. More qualitative analysis of CoinSeg. The white borderline is **ignore label** in Pascal VOC 2012 dataset.

References

[1] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulo, Elisa Ricci, and Barbara Caputo. Modeling the background for incremental learning in semantic segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 9233–9242, 2020. 1, 2, 3

[2] Sungmin Cha, YoungJoon Yoo, Taesup Moon, et al. Ssul: Semantic segmentation with unknown label for exemplar-based class-incremental learning. *Advances in neural information processing systems*, 34, 2021. 1, 2, 3

[3] Arthur Douillard, Yifu Chen, Arnaud Dapogny, and Matthieu Cord. Plop: Learning without forgetting for continual semantic segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4050, 2021. 2, 3

[4] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal on Computer Vision*, 88(2):303–338, 2010. 1

[5] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on vision transformer. *IEEE transactions on pattern analysis and machine intelligence*, 45(1):87–110, 2022. 3

[6] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 40(12):2935–2947, 2017. 2, 3

[7] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 1, 3

[8] Umberto Michieli and Pietro Zanuttigh. Incremental learning techniques for semantic segmentation. In *Proc. IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2, 3

[9] Umberto Michieli and Pietro Zanuttigh. Continual semantic segmentation via repulsion-attraction of sparse and disentangled latent representations. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1114–1124, 2021. 1, 2, 3

[10] Chang-Bin Zhang, Jia-Wen Xiao, Xialei Liu, Ying-Cong Chen, and Ming-Ming Cheng. Representation compensation networks for continual semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7053–7064, 2022. 1, 2

[11] Zekang Zhang, Guangyu Gao, Zhiyuan Fang, Jianbo Jiao, and Yunchao Wei. Mining unseen classes via regional objectness: A simple baseline for incremental segmentation. *Advances in neural information processing systems*, 35, 2022. 1, 2, 3

[12] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 633–641, 2017. 4