Exploiting a Joint Embedding Space for Generalized Zero-Shot Semantic Segmentation Supplement

Donghyeon Baek* Youngmin Oh* Bumsub Ham†
School of Electrical and Electronic Engineering, Yonsei University
https://cvlab.yonsei.ac.kr/projects/JoEm

Here we present a detailed description of the Apollonius circle in Sec. 1, and show more results on different experiment settings in Sec. 2.

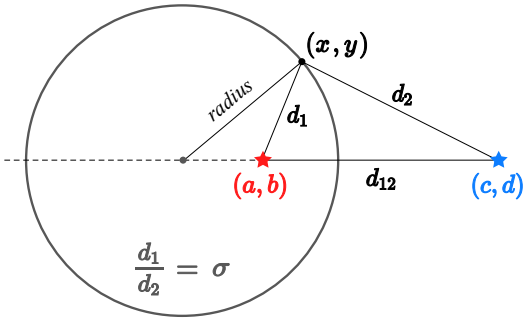


Figure 1. Illustration of the Apollonius circle in a two-dimensional Euclidean space.

1. Apollonius' definition of a circle

A circle in a Euclidean space refers to a set of points, where all distances from each point to a particular point (*i.e.*, *center*) are equal. Apollonius of Perga, on the other hand, defines a circle as a set of points that have the same distance ratio for two particular points (Fig. 1). Concretely, the Apollonius circle in a two-dimensional Euclidean space is defined as follows:

$$\begin{aligned}
&\frac{d_1}{d_2} = \sigma \\
&\iff \frac{\sqrt{(x-a)^2+(y-b)^2}}{\sqrt{(x-c)^2+(y-d)^2}} = \sigma \\
&\iff (x-a)^2+(y-b)^2 = \sigma^2\left((x-c)^2+(y-d)^2\right) \\
&\iff \left(x-\frac{a-\sigma^2 c}{1-\sigma^2}\right)^2+\left(y-\frac{b-\sigma^2 d}{1-\sigma^2}\right)^2 = \frac{\sigma^2}{(1-\sigma^2)^2}\left((a-c)^2+(b-d)^2\right),
\end{aligned} \tag{1}$$

where (a, b) and (c, d) are the two particular points, often called *foci*. From the last row in Eq. (1), we reformulate the radius of this circle as follows:

$$\begin{aligned}
radius &= \frac{\sigma}{1-\sigma^2}\sqrt{(a-c)^2+(b-d)^2} \\
&= \frac{\sigma}{1-\sigma^2} d_{12}
\end{aligned} \tag{2}$$

We can see that the radius is proportional to the distance between the *foci*, *i.e.*, d_{12}. This confirms that Apollonius calibration handles the seen bias problem adaptively, even with the same value of σ (see Figure 4 in the main paper).

*Equal contribution, †Corresponding author.

Table 1. Comparison of different experiment settings on PASCAL VOC [6]. Exclude(*classes*): exclude all samples that contain at least one of the *classes*; Ignore(*classes*): mark regions of the *classes* in ground-truth masks with *void* labels; IN: pre-trained weights for ImageNet classification [5]; IN*: pre-trained weights for ImageNet classification only with seen classes.

Settings	# of splits	Training	Inference	Architecture (backbone)	Initialization	Semantic features
ZS3Net [2]	5	Exclude(unseen classes)	-	DeepLabV3+ [4] (ResNet-101 [8])	IN*	*word2vec* [10]
SPNet [12]	1	Ignore(background & unseen classes)	Ignore(background class)	DeepLabV2 [3] (ResNet-101 [8])	IN	*word2vec* [10] & *fastText* [9]

2. More results

Table 1 summarizes differences between experiment settings provided by ZS3Net [2] and SPNet [12] on PASCAL VOC [6]. The main differences between them lie in training and inference processes. ZS3Net excludes training samples that contain at least one of the unseen classes. This, however, would leave a small number of training samples only, since objects co-occur frequently. For example, almost half of the training samples are eliminated for the unseen-10 split ($10,582 \rightarrow 5,408$). To take into account this problem, SPNet uses all training samples but ignores pixels of unseen classes, which is more feasible for the task of semantic segmentation. Note that SPNet however ignores the background class during both training and inference. That is, SPNet requires pixel-wise annotations for the background class to discriminate foreground objects from a background during inference, while ZS3Net does not impose any assumptions for inference. This is why we have followed the setting of ZS3Net in the main paper. In the following, we present qualitative and quantitative results for each experiment setting.

2.1. ZS3Net.

We vary the value of r to analyze its effect on the unseen-4 split of PASCAL VOC [6] and Context [11] in Fig. 2. We can see that using $r > 1$ always gives better results than $r = 1$, confirming again the superiority of our BAR loss. We empirically set r to 4 in all experimental settings provided by ZS3Net [2] in order to generate more virtual prototypes. Since the spatial size of feature maps obtained from DeepLabV3+ is already small (*e.g.*, 78×78), we do not use higher values of r, *i.e.*, $r > 4$, to maintain the spatial size. For temperatures in Eqs. (7) and (8), we first fix τ_s as one in order to reduce the cost of selecting these hyperparmeters. Then, we find a value of τ_μ that allows the highest relation probability in Eq. (8) to be around 0.9. As a result, we empirically set τ_μ to 5 and 7 for all experiments on PASCAL VOC and Context, respectively. We also analyze effects of other hyperparameters (λ, σ) on PASCAL VOC and Context in Figs. 3 and 4, respectively. We first use a grid search to set $\lambda \in \{10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$ with fixing the adjustable parameter σ to 0.5. From the top rows of Figs. 3 and 4, we can clearly see that our approach is robust to changes of the balance parameter λ. We then vary σ in the range of $(0, 1)$ with an interval of 0.1. From the bottom rows of Figs. 3 and 4, we can also see that Apollonius calibration brings considerable performance improvement in the range of $[0.5, 1)$. This method, however, degrades performance in the range of $(0, 0.5)$. A plausible reason is that the radius of the Apollonius circle becomes small as in Eq. (2), increasing classification errors. It is worth noting that the values of hyperparameter used in the main paper (dotted lines) are not optimal, since we have adopted a cross-validation [1] for each split.

We compare in Table 2 per-class mIoU scores on the unseen-4 split of PASCAL VOC. We can see that ZS3Net [2] obtains mIoU$_\mathcal{U}$ scores at the cost of mIoU$_\mathcal{S}$ ones. For example, the mIoU score of a sheep class is 0. We reimplement ZS3Net ('ZS3Net†'), and find that ZS3Net requires a scaling factor for unseen classes to compute the CE loss during retraining. This scaling factor is particularly important for improving performance in that ZS3Net† outperforms ZS3Net by simply changing the value of the scaling factor from 100 to 10. We also report per-class results of ZS3Net using our visual encoder ('ZS3Net‡'). This model shows higher mIoU$_\mathcal{U}$ scores than ZS3Net† for cow, motorbike, and sofa classes, demonstrating once again that our approach alleviates the seen bias problem (see Table 3 in the main paper).

We provide in Fig. 5 visual examples using different losses in our framework. To the baseline, we show results without BAR and SC losses in the third column. We can see that BAR and SC losses give better results than the baseline in the fourth and fifth columns, respectively. In the two rightmost columns, we show BAR and SC losses complement each other and Apollonius calibration considerably reduces the seen bias problem, respectively. In Figs. 7 and 8, we present qualitative results on the unseen-4 split of PASCAL VOC and Context, respectively. We can see that our approach provides more accurate results than ZS3Net, especially for unseen classes.

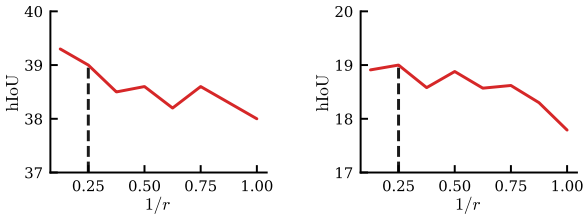
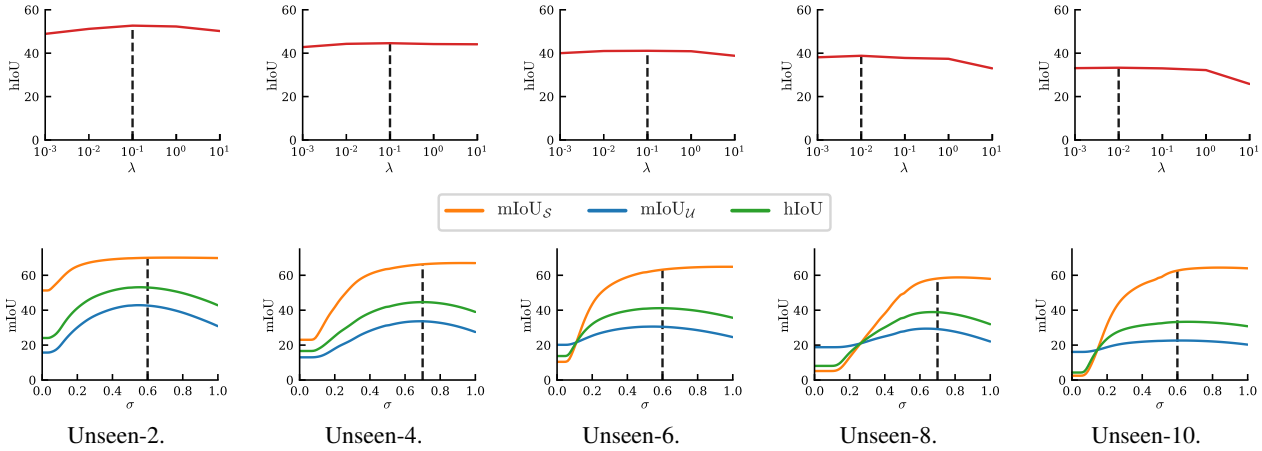
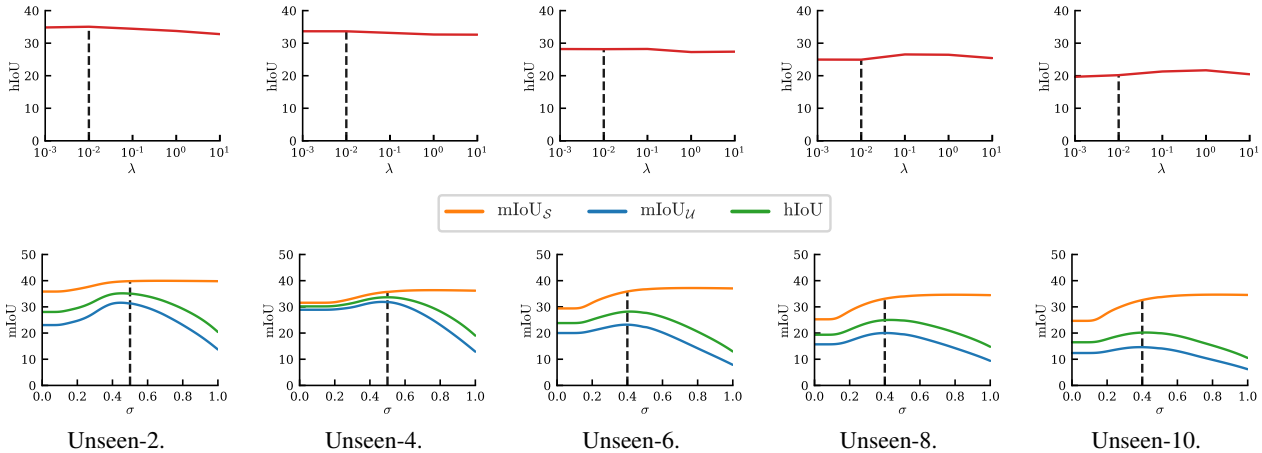


Figure 2. Comparison of hIoU scores for different values of r on the unseen-4 split of PASCAL VOC [6] (left) and PASCAL Context [11] (right). The dotted line indicates the value used in the main paper.



Unseen-2. Unseen-4. Unseen-6. Unseen-8. Unseen-10.

Figure 3. Comparison of mIoU scores for different hyperparameters λ (top) and σ (bottom) on each split of PASCAL VOC [6]. The dotted line indicates the value used in the main paper.



Unseen-2. Unseen-4. Unseen-6. Unseen-8. Unseen-10.

Figure 4. Comparison of mIoU scores for different hyperparameters λ (top) and σ (bottom) on each split of PASCAL Context [11]. The dotted line indicates the value used in the main paper.

Table 2. Per-class mIoU scores on the unseen-4 split of PASCAL VOC [6]. *: unseen classes; †: reimplementation; ‡: our visual encoder. The results of ZS3Net [2] are obtained from the model provided by the authors, but differ from the original ones. Numbers in bold are the best performance and underlined one is the second best.

Method	bg.	aero*	bike	bird	boat	bot	bus	car	cat	cha	cow*	tab	dog	hor	mbik*	pers	plnt	she	sofa*	trai	tv	hIoU
ZS3Net [2]	**92.2**	32.8	**35.7**	85.2	56.1	**78.9**	91.0	77.5	84.3	**26.2**	27.3	41.9	80.4	**57.2**	39.5	**81.5**	**58.8**	0.0	9.1	79.5	**69.3**	38.3
ZS3Net†	91.8	**37.1**	32.2	**85.8**	57.2	75.8	**91.5**	71.8	**89.8**	24.8	30.4	49.8	**86.6**	51.7	43.0	81.1	55.6	72.2	4.5	84.2	67.7	40.6
ZS3Net‡	91.0	35.6	32.1	85.5	58.3	73.9	90.0	**77.6**	86.4	25.0	**31.7**	**53.9**	83.1	53.0	48.3	80.5	51.5	**73.0**	11.7	**85.3**	62.4	43.4
Ours	90.2	36.2	29.6	83.5	**60.4**	69.7	90.3	76.8	87.8	18.6	30.1	49.8	83.8	52.1	**52.9**	78.1	51.2	58.4	**15.3**	84.8	62.2	**44.6**

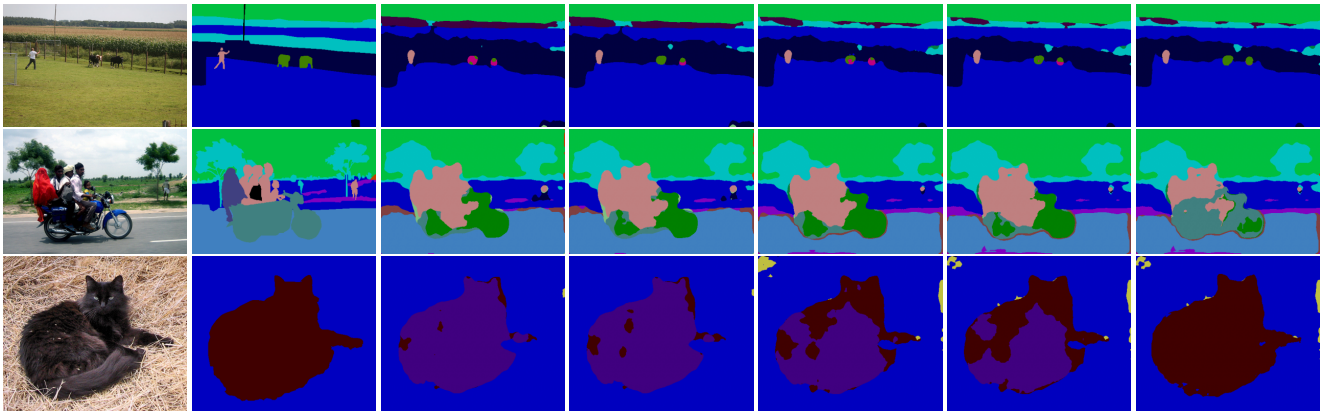


Figure 5. Visual comparison using different losses in our framework on the unseen-4 split of PASCAL Context [11]. Note that cow, motorbike, and cat are unseen classes.

2.2. SPNet.

To further demonstrate the effectiveness of our approach, we follow the experiment setting provided by SPNet [12] on PASCAL VOC [6]. Specifically, it provides a single split that consists of 15 seen and 5 unseen classes (potted-plant, sheep, sofa, train, and tv). For fair comparison, we use DeepLabV2 [3] with ResNet-101 [8] as our visual encoder. ResNet-101 is initialized by pre-trained weights for ImageNet classification [5]. For side information, we concatenate *word2vec* [10] and *fastText* [9], resulting in a 600-dimensional semantic feature. We use the same training details aforementioned in the main paper to train both encoders.

We have found that many virtual prototypes are located at boundaries between foreground and background. As the setting provided by SPNet ignores background regions, we set r to 2. For temperature parameters, we adopt the same values as in the experimental settings provided by ZS3Net. Other hyperparameters ($\lambda = 1, \sigma = 0.8$) are chosen by a cross-validation as in [1]. Fig. 6 shows performance variations w.r.t. the value of r and σ. We can see that the behavior of mIoU scores w.r.t. these hyperparameters is similar to the ones in Figs. 3 and 4.

We compare in Table 3 our approach with state-of-the-art GZS3 methods [2, 7, 12]. All numbers for other methods are taken from CaGNet [7]. From this table, we can clearly see that our approach outperforms all other methods including generative methods [2, 7] by a large margin in terms of mIoU$_{\mathcal{U}}$ and hIoU. Note that our approach without using AC already outperforms all other methods, demonstrating the effectiveness of the discriminative approach. In Fig. 9, we show qualitative comparison of ours and CaGNet. We can clearly see that our approach provides better results.

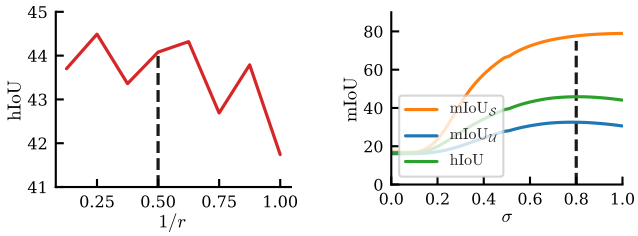


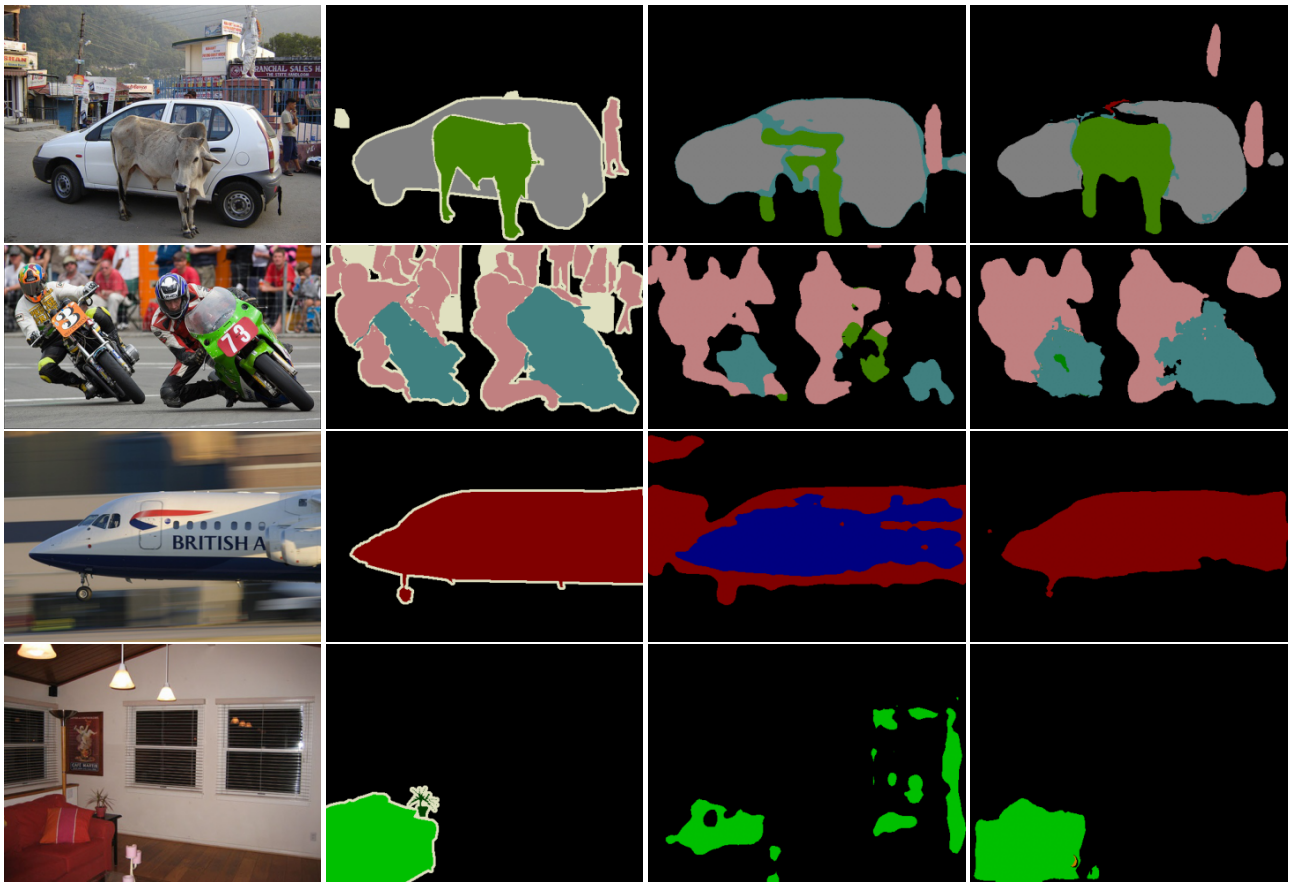
Figure 6. Comparison of mIoU scores for different hyperparameters r (left) and σ (right) on the experiment setting provided by SPNet [12]. We empirically set λ to 1. The dotted line indicates the value chosen by a cross-validation.

References

[1] Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. Generating visual representations for zero-shot classification. In *ICCV Workshops*, 2017. 2, 4

Table 3. Quantitative comparison of state-of-the-art methods [2, 7, 12] and ours on the experiment setting provided by SPNet [12] in terms of mIoU.

Methods	$\text{mIoU}_{\mathcal{S}}$	$\text{mIoU}_{\mathcal{U}}$	hIoU
SPNet [12]	78.0	15.6	26.1
ZS3Net [2]	77.3	17.7	28.7
CaGNet [7]	78.4	26.6	39.7
Ours w/o AC	78.9	30.6	<u>44.1</u>
Ours	77.7	32.5	**45.9**



Input image. Ground truth. ZS3Net. Ours.

Figure 7. Visual comparison of ZS3Net [2] and ours on the unseen-4 split of PASCAL VOC [6]. Note that cow, motorbike, airplane, and sofa are unseen classes.

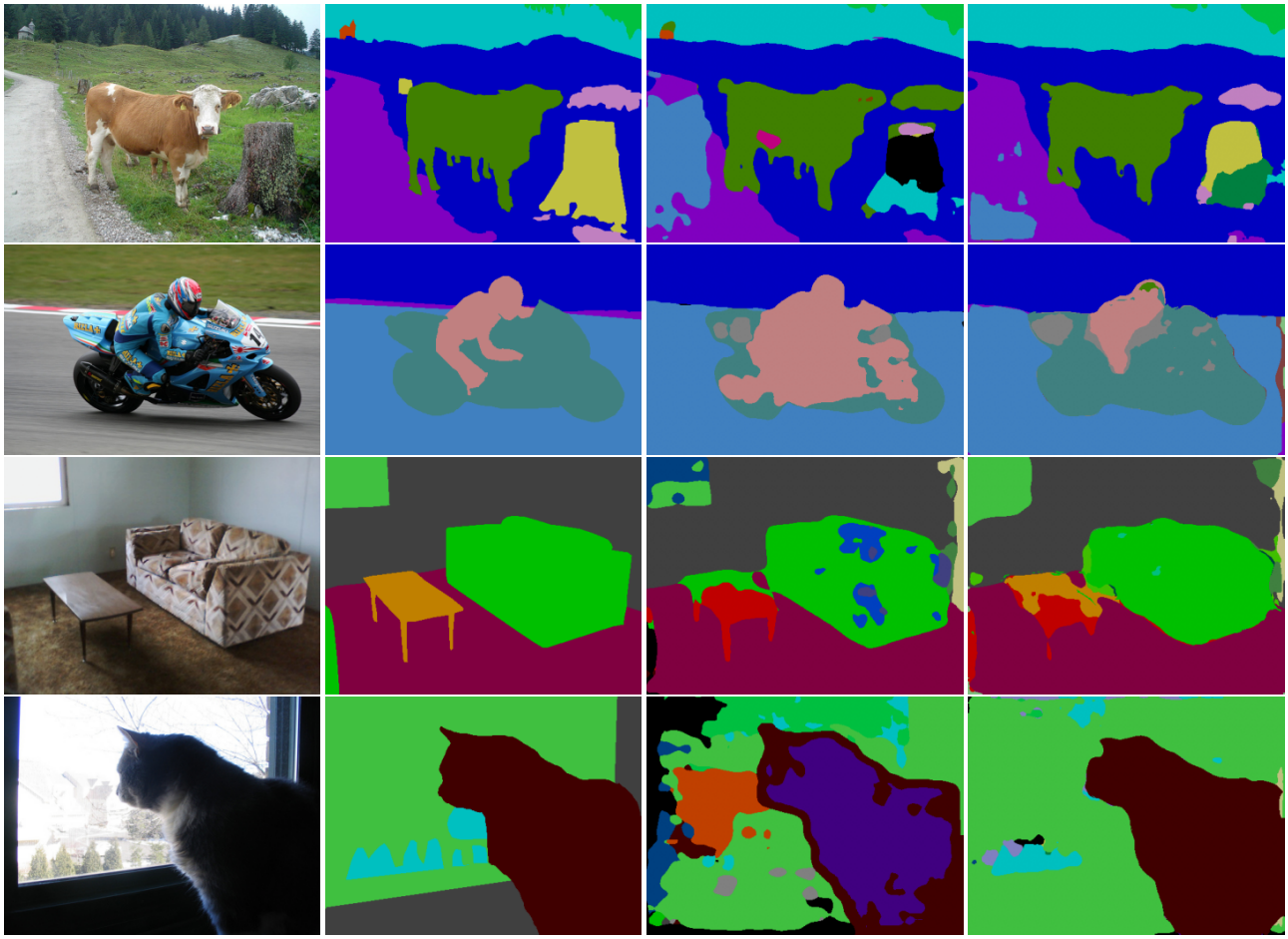
[2] Maxime Bucher, VU Tuan-Hung, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. In *NeurIPS*, 2019. 2, 3, 4, 5, 6

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. PAMI*, 40(4):834–848, 2017. 2, 4

[4] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 2

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 2, 4

[6] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010. 2, 3, 4, 5



Input image. Ground truth. ZS3Net. Ours.

Figure 8. Visual comparison of ZS3Net [2] and ours on the unseen-4 split of PASCAL Context [11]. Note that cow, motorbike, sofa, and cat are unseen classes.

[7] Zhangxuan Gu, Siyuan Zhou, Li Niu, Zihan Zhao, and Liqing Zhang. Context-aware feature generation for zero-shot semantic segmentation. In *ACM MM*, 2020. 4, 5, 7

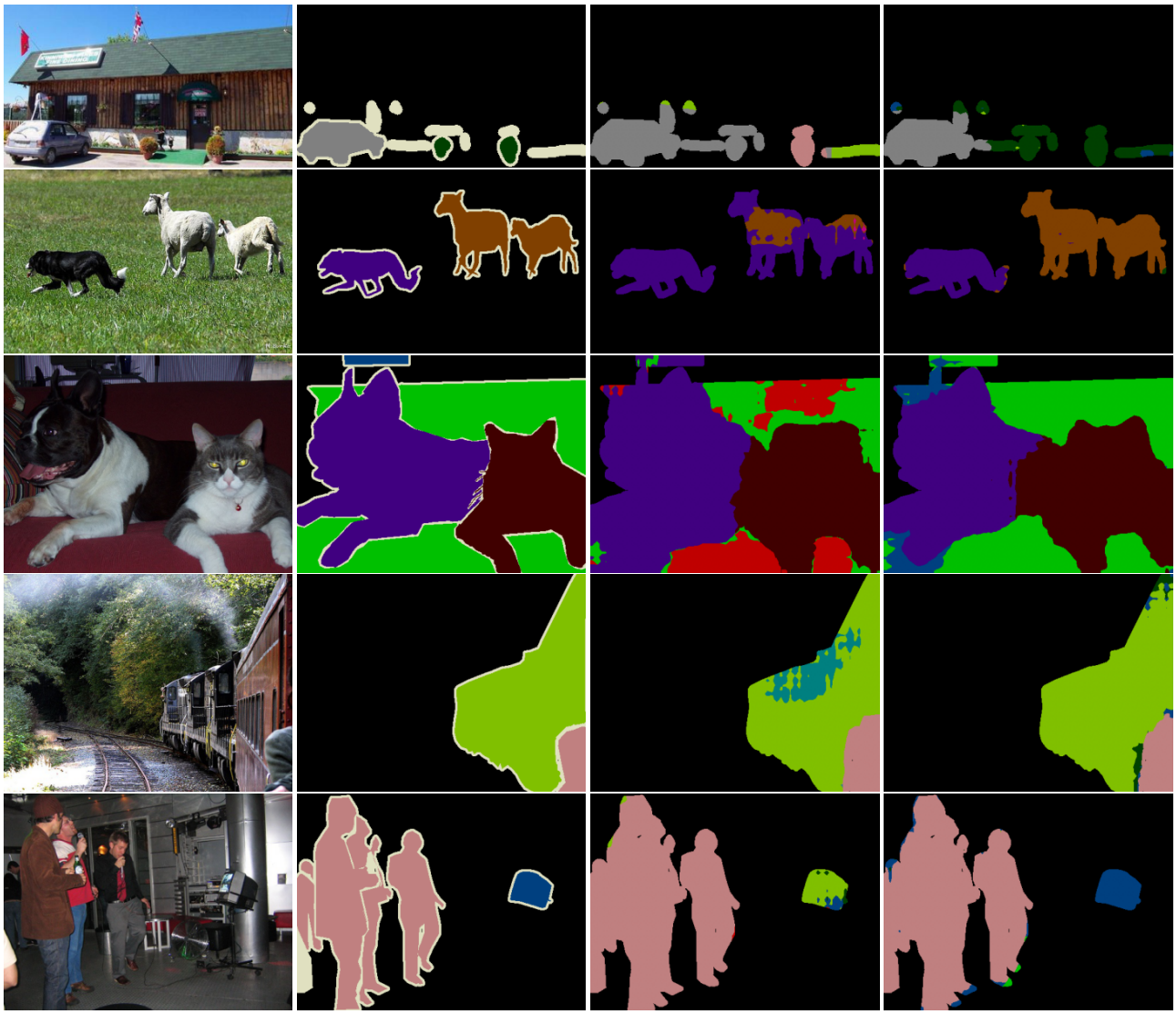
[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2, 4

[9] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. FastText.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016. 2, 4

[10] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *NeurIPS*, 2013. 2, 4

[11] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 2014. 2, 3, 4, 6

[12] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *CVPR*, 2019. 2, 4, 5, 7



Ground truth. Ground truth. CaGNet. Ours.

Figure 9. Visual comparison of CaGNet [7] and ours on the experiment setting provided by SPNet [12], where potted-plant, sheep, sofa, train, and tv are unseen classes. Note that all results show sharp object boundaries, since this setting uses pixel-wise annotations for the background class during inference.