Supplementary Materials for *Prototypical Kernel Learning and Open-set Foreground Perception for Generalized Few-shot Semantic Segmentation*

Kai Huang, Feigege Wang, Ye Xi, Yutao Gao
Alibaba Group
zhouwan.hk, feigege.wfgg, yx150449, yutao.gao@alibaba-inc.com

1. Limitation Statement

As shown in Table 1 and Table 5 of main body, the performance gain on COCO-20^i is apparent less than PASCAL-5^i with the speculation that simple feature aggregation, *i.e*, pixel feature assembling, can not handle the complex prototype relation comparison as the number of classes increases. In the light of this limitation, one of the possible future work is proceed to replace the straightforward overall feature pooling with better instance-level knowledge aggregation strategies for more distinguishable class representation. And dense interaction such as transformer has been proved to be more robust in feature comparison of few-shot scenario [6], which has the potential for improving the performance of GFSS model as well.

2. Implementation Details

Following [4], we perform the multi-fold cross validation on PASCAL-5^i and COCO-20^i datasets for credible evaluation. Take the PASCAL-5^i for example, the whole 20 target classes are average divided into four folds. For a specific fold, the target classes of this fold are treated as novel classes, in which only few annotated data is available, *e.g.*, 1-shot or 5-shot. The rest target classes of other folds as well as background class are constituted as base classes set. Different from the novel classes set, the labeled data of base classes are adequate. In training process, the model can both access to the base and novel classes. And in evaluation process, we should test our model on the whole val set of PASCAL which contains base and novel classes simultaneously. There is no additional information, such as support data in normal FSS, will be provided. As for CIFSS, we continue the multi-fold cross validation on PASCAL-5^i and COCO-20^i, in which same fold partition is adopted. For the selected specific fold, the target classes are further evenly split into several parts, *i.e.*, novel sessions. Likewise, all the samples of base classes are built up the base session. The novel sessions are processed with a incremental stream, and usually high consumption approaches are not allowed, such as store those incremental samples or feature maps. Lightweight memory module or feature prototypes are alternative options.

In addition to the evaluation metric of overall mIoU, we also adopt the hIoU metric for more comprehensive comparison. The harmonic mean of base-class mIoU $m\text{IoU}_{\mathcal{B}}$ and novel-class mIoU $m\text{IoU}_{\mathcal{N}}$ is calculated by:

$$h\text{IoU} = \frac{2 \times m\text{IoU}_{\mathcal{B}} \times m\text{IoU}_{\mathcal{N}}}{m\text{IoU}_{\mathcal{B}} + m\text{IoU}_{\mathcal{N}}}. \tag{1}$$

The the number and performance of base classes are commonly higher than novel classes, leading to the domination of overall *m*IoU. The *h*IoU thus can better demonstrate the performance of method.

3. Additional Results and Analyses for GFSS

Detailed Numerical Results for Each Fold. Table 1 and Table 2 show the 1-shot and 5-shot detailed performance comparisons specific to each fold respectively. It obvious that our approach outperforms other methods with significant improvement, which further demonstrates the effectiveness of the proposed method. Beside, the substantial gain on all folds of both PASCAL-5^i and COCO-20^i indicates the robustness of the proposed method with different partitions of base and novel classes set.

Performing Prototypical Update on Just Base-class Kernels or Whole-class Kernels. In the conditional bias based inference, we only update the base-class kernels with the PKL module for keeping the consistency with training process. And another consideration is to mitigate the noise accumulation of novel-class kernels for the authentic representation of novel targets. As shown in Table 3, we perform the whole-class kernels update with the PKL module which is indicated as *whole*CBBI, and the original base-class kernels update is named as *base*CBBI. It is clear that the prototypical update for novel-class kernels brings adverse impact, which are aggregated with few valid data and instability with noise sensitive, leading to worse performance in both novel classes set and base classes set.

Can Prototypical Update be Plug and Play? The PKL module is designed to refine the base-class kernels with spe-

Table 1. The detailed performance comparison of 1-shot generalized few-shot semantic segmentation on PASCAL-5^i and COCO-20^i. Best-performing results are highlighted in bold.

method	fold-0				fold-1				fold-2				fold-3			
	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$
									PASCAL-5^i							
BAM [2]	68.43	9.90	54.49	17.30	61.17	24.43	52.42	34.92	60.94	21.14	51.46	31.39	68.06	12.77	54.90	21.51
CAPL [4]	69.72	11.47	55.85	19.70	63.02	25.95	54.19	36.76	61.41	20.35	51.64	30.57	70.20	12.04	56.35	20.55
Ours	**72.32**	**24.56**	**60.95**	**36.67**	**64.56**	**38.97**	**58.47**	**48.60**	**64.90**	**26.32**	**55.71**	**37.45**	**73.58**	**17.76**	**60.29**	**28.61**
									COCO-20^i							
BAM* [2]	37.50	3.39	29.07	6.15	44.07	8.84	35.37	14.73	45.83	4.41	35.60	8.05	45.11	7.70	35.87	13.15
CAPL* [4]	39.73	5.26	31.20	9.29	45.15	10.12	36.50	16.53	48.28	6.82	38.04	11.95	46.90	8.36	37.38	14.19
Ours	**42.06**	**8.43**	**33.76**	**14.04**	**46.85**	**12.89**	**38.76**	**20.22**	**49.10**	**10.40**	**39.55**	**17.16**	**47.41**	**12.44**	**38.77**	**19.71**

Table 2. The detailed performance comparison of 5-shot generalized few-shot semantic segmentation on PASCAL-5^i and COCO-20^i. Best-performing results are highlighted in bold.

method	fold-0				fold-1				fold-2				fold-3			
	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	novel	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$
									PASCAL-5^i							
BAM [2]	67.73	10.32	54.01	7.91	62.45	23.39	53.15	34.03	60.84	18.72	50.81	28.63	68.18	4.32	52.98	8.13
CAPL [4]	69.72	11.47	55.85	19.70	63.02	25.95	54.19	36.76	61.41	20.35	51.64	30.57	70.20	12.04	56.35	20.55
Ours	**72.47**	**29.32**	**62.20**	**41.75**	**67.25**	**46.34**	**62.28**	**54.87**	**63.37**	**32.56**	**56.03**	**43.02**	**73.77**	**29.38**	**63.20**	**42.02**
									COCO-20^i							
BAM* [2]	38.43	4.73	30.11	8.42	43.28	8.59	34.71	14.33	48.03	6.31	37.73	11.15	45.51	7.84	36.21	13.38
CAPL* [4]	40.14	6.81	31.91	11.64	43.79	8.26	35.02	13.90	45.71	5.23	35.71	17.77	46.39	9.14	37.19	15.27
Ours	**42.61**	**10.79**	**34.75**	**16.82**	**47.05**	**16.77**	**39.57**	**24.73**	**48.91**	**15.88**	**40.75**	**23.96**	**47.39**	**16.95**	**39.87**	**24.97**

Table 3. Quantitative results with different prototype updating setting on the inference of PASCAL-5^i and COCO-20^i. Best-performing results are highlighted in bold.

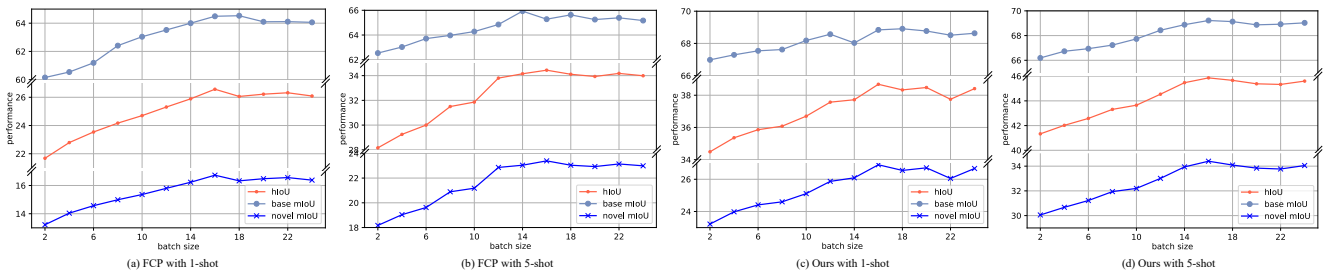
Methods	1-shot				5-shot			
	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$m\text{IoU}_{\mathcal{O}}$	$h\text{IoU}$
				PASCAL-5^i				
*whole*CBBI	65.58	20.22	54.78	30.91	66.84	26.40	57.21	37.85
*base*CBBI	**68.84**	**26.90**	**58.86**	**37.83**	**69.22**	**34.40**	**61.18**	**45.42**
				COCO-20^i				
*whole*CBBI	44.91	8.74	35.98	14.63	45.26	11.93	37.03	18.88
*base*CBBI	**46.36**	**11.04**	**37.71**	**17.83**	**46.77**	**14.91**	**38.90**	**22.61**

Table 4. Ablation performances of the PKL module on PASCAL-5^i.

Methods	1-shot			5-shot		
	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$h\text{IoU}$
BAM [2]	64.65	17.06	27.00	65.28	19.99	30.61
BAM*	64.32	15.89	25.48	64.90	18.07	28.27
BAM* *w/ pp*PKL	63.51	12.29	20.59	64.18	16.76	26.58
BAM* *w/* PKL	65.58	20.45	31.18	66.37	23.39	34.59
Ours *w/ pp*PKL	65.83	22.17	33.17	66.64	26.08	37.49
Ours	**68.84**	**26.90**	**37.83**	**69.22**	**34.40**	**45.42**

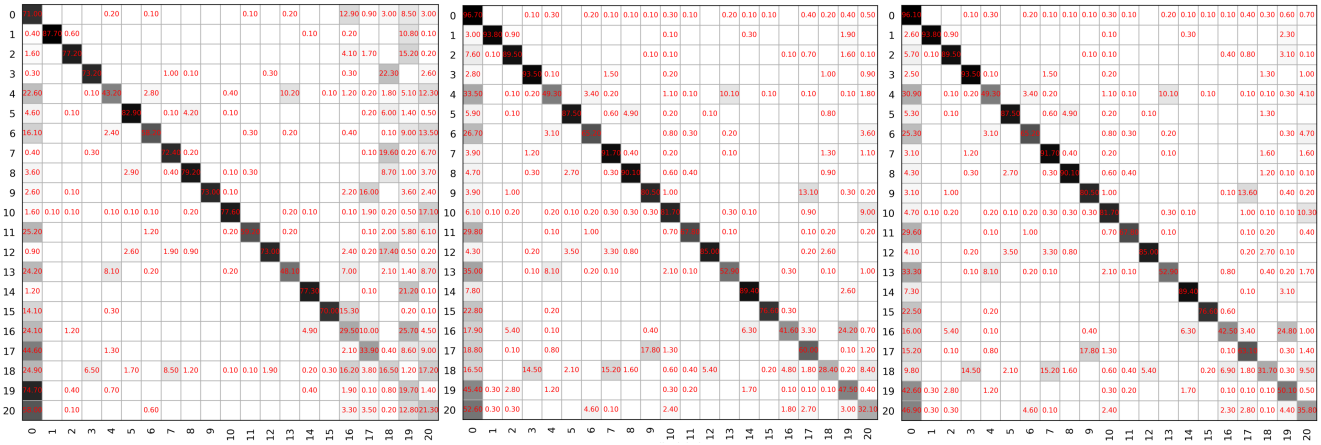
cific input images. Table 4 explores whether it can be a plug and play module to the prototype-based methods. We retrain the base learner of BAM [2] with the prototypical kernels, and then optimize the meta learner following the original setting, which is represented as BAM*. The PKL with the format of plug and play, indicated as *pp*PKL, is directly adopted in the inference process of BAM* and the proposed method (without PKL in training process), corresponded to BAM* *w/ pp*PKL and ours *w/ pp*PKL respectively. We can see that (i) the modified BAM [2] with prototypical kernels can achieve similar results the same as original BAM, and the PKL module improves the prototypical BAM with a significant gain as well; (ii) the PKL module with plug and play format deteriorates the performance of prototypical BAM and the proposed method simultaneously, demonstrating that it can not directly be a plug and play module. A plain explanation is that the PKL serves to optimize the feature representation for stronger as well as more stable prototypical update in the training process.

The Influence of Batch Size for Foreground Contextual Perception Module. The pseudo episode mechanism cooperating with FCP module is proposed to strengthen the foreground perception with contextual targets, thus the number of images as well as batch size in each pseudo episode is a important parameter. We take the batch size from 2 to 24 with step 2 to study its influence. The metrics of $m\text{IoU}_{\mathcal{B}}$, $m\text{IoU}_{\mathcal{N}}$ and $h\text{IoU}$ on the PASCAL-5^i benchmark of 1-shot are used for illustration. As shown in Fig. 1, the performance is continuously improved as the batch size increases and up to the best results when the batch size is set as 8, which is the final value used in all our experiments. Since we perform the single forward in the inference process, we also perform the batch-based inference, which obtains similar results with the fluctuation of 1%. It is clear that the batch size only slightly affects performance, thus the model can get rid of the restriction of batch input after finishing



(a) FCP with 1-shot (b) FCP with 5-shot (c) Ours with 1-shot (d) Ours with 5-shot

Figure 1. Ablation studies on the batch size of FCP module. The model of (a) and (b) corresponds to the third line of Table 3.



(a) Baseline (b) Baseline *w*/ MAU (c) Ours

Figure 2. Visual results of confusion matrix. The horizontal axis means the predicted labels and the vertical axis indicates the ground truth. The "Baseline" of (a) is the model of first line in Table 3, and the "Baseline *w*/ PKL" of (b) corresponds to the model of second line in Table 3.

the training process.

Pixel-wise Correlation Map *vs.* Refined Correlation Prototypes. We further test these two way for cross targets correlation expression, the model with refined correlation response superior the way of pixel-wise correlation map by 1.85% and 3.30% in term of $m\text{IoU}_{\mathcal{O}}$ and *h*IoU respectively. Compared with pixel-wise correlation map, the refined correlation response is more robust with fewer pixel sharp noises and concentrates on the categorical attributes for more general foreground perception.

Model Effectiveness from the View of Confusion Matrix. Fig. 2 visualizes the overall confusion matrix of PASCAL-5^i fold-0, the class id from 0 to 15 represent the base classes, and the rest class id belong to novel classes. It is obvious that the baseline model suffer from the representation division and embedding prejudice, in which the targets of novel classes are misidentified as base classes and background respectively. After introducing PKL module, the chaos of novel classes confusion matrix (five columns on the right) is refreshed and the correct recognized rate of novel classes are increased (on the diagonal). Furthermore, our model by leveraging the FCP module for open-set foreground perception can offer more accurate generalized segmentation, further mitigating the interference of background (the last five row in first column).

Comparison of Parameter Quantity and Time Consumption. Table 6 shows the learnable parameter quantity and time consumption of 1-shot inference compared with other approaches. Although the proposed method requires modular processing and output assembling, our method are friendly to the parameter and time consumption of model with flexible as well as concise structures, *e.g.*, class-wise kernels and lightweight decoder. Moreover, most of these operations, especially for the conditional bias based inference module, can be accelerated through matrix computations.

4. Additional Results and Analyses for CIFSS

Performance Comparison on 5-shot Setting. Table 7 presents the quantitative results of 5-shot class incremental few-shot semantic segmentation on PASCAL-5^i and COCO-20^i. Similar to the conclusion of 1-shot setting, our method significantly outperforms recent methods in term of mIoUn and hIoU. And as the novel session increases, the performance superiority of the proposed method is constantly strengthened.

Table 5. The performance comparison of 5-shot class incremental few-shot semantic segmentation on PASCAL-5^i and COCO-20^i. Best-performing results are highlighted in bold.

Datasets	Methods	sesson 0	session 1			session 2			session 3			session 4			session 5		
		$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$h\text{IoU}$
PASCAL-5^i	PFENet [5]	74.43	68.80	18.82	29.56	66.61	23.30	34.52	63.91	17.77	27.81	60.85	21.12	31.36	58.78	15.65	24.72
	iFS-RCNN [3]	72.43	70.49	22.73	34.38	68.81	25.11	36.79	67.74	21.15	32.24	64.03	25.50	36.47	62.42	19.94	30.22
	CAPL [4]	74.86	71.19	22.83	34.57	68.88	24.49	36.13	68.08	24.07	35.57	64.89	26.99	38.12	63.21	20.84	31.35
	BAM [2]	**75.83**	**74.88**	24.41	36.82	70.91	26.73	38.82	68.49	25.55	37.22	65.36	27.71	38.92	63.34	22.58	33.29
	PIFS [1]	75.04	73.41	23.39	35.48	68.82	24.43	36.06	68.50	26.76	38.49	66.61	28.44	39.86	63.59	26.63	37.54
	Our	75.49	73.00	**30.64**	**43.16**	**71.52**	**34.98**	**46.98**	**71.09**	**33.06**	**45.13**	**70.37**	**33.13**	**45.05**	**68.89**	**34.56**	**46.03**
COCO-20^i	PFENet [5]	54.11	48.92	8.81	14.93	44.30	10.07	16.41	40.47	6.64	11.41	39.95	8.88	14.53	38.71	9.90	15.77
	iFS-RCNN [3]	53.42	52.28	10.84	17.96	46.69	12.11	19.23	43.77	8.12	13.70	41.01	10.10	16.21	40.06	11.73	18.15
	CAPL [4]	54.43	52.90	11.69	19.15	47.18	12.29	19.50	44.43	8.86	14.77	41.39	11.47	17.9	40.41	12.22	18.77
	BAM [2]	**54.80**	**52.73**	14.47	22.71	**48.60**	14.43	22.25	46.72	11.19	18.06	44.96	14.08	21.44	43.35	14.88	22.16
	PIFS [1]	54.27	51.95	14.68	22.89	48.04	13.32	20.86	46.93	11.82	18.88	45.08	14.66	22.12	43.57	14.13	21.34
	Our	54.39	51.82	**17.00**	**25.60**	48.33	**16.65**	**24.77**	**47.10**	**14.17**	**21.79**	**46.16**	**16.67**	**24.49**	**45.22**	**16.02**	**23.66**

Table 6. Comparison of parameter quantity and time consumption on 1-shot setting during inference stage.

method	Learnable Parameter	Speed
PFENet [5]	10.8M	12.44 FPS
BAM [2]	26.7M	7.49 FPS
CAPL [4]	17.4M	10.20 FPS
Our	18.1M	9.35 FPS

Table 7. Ablation Study of the effect with different components in CIFSS setting.

PKL	FCP	CBBI	1-shot			5-shot		
			$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$h\text{IoU}$	$m\text{IoU}_{\mathcal{B}}$	$m\text{IoU}_{\mathcal{N}}$	$h\text{IoU}$
			64.39	11.23	19.12	65.51	16.90	26.87
✓			68.54	21.10	32.27	69.25	25.77	37.56
	✓		66.47	17.82	28.11	67.89	23.06	34.43
✓	✓		70.04	22.91	34.53	70.43	28.81	40.89
✓	✓	✓	**70.66**	**26.61**	**38.66**	**70.90**	**33.27**	**45.31**

Ablation Results of the Effect with Different Components. We also take the ablation studies of different components with the CIFSS scenario shown in Table 7. Compared to the performance of the proposed method, the model without the meta-prototype adaptive updating module, foreground contextual perception module, and conditional bias based inference descends it to 10.55%, 6.39%, and 4.13% of hIoU, respectively. These results demonstrate that the two proposed modules, PKL module and FCP module, have more impact on performance improvement of CIFSS. And the combination of the two modules with CBBI leads to our method accomplishing the highest performance, which further proves the effectiveness of the proposed method.

5. Additional Qualitative Results

In this section, we present more qualitative results of the BAM [2], CAPL [4], and the proposed method compared to demonstrate its generalized few-shot segmentation performance. Appearance and scale variations are the innate difficulty of the generalized few-shot semantic segmentation task. The examples of PASCAL-5^i benchmark are shown in Fig. 3, our model exhibits great superiority in alleviating appearance and scale variations. Besides, we also sample some examples from coco-20^i benchmark, and the qualitative results are presented in Fig. 4.

References

[1] Fabio Cermelli, Massimiliano Mancini, Yongqin Xian, Zeynep Akata, and Barbara Caputo. Prototype-based incremental few-shot semantic segmentation. In *British Machine Vision Conference*, 2021. 4

[2] Chunbo Lang, Gong Cheng, Binfei Tu, and Junwei Han. Learning what not to segment: A new perspective on few-shot segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8057–8067, 2022. 2, 4, 5, 6

[3] Khoi Nguyen and Sinisa Todorovic. ifs-rcnn: An incremental few-shot instance segmenter. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7010–7019, 2022. 4

[4] Zhuotao Tian, Xin Lai, Li Jiang, Shu Liu, Michelle Shu, Hengshuang Zhao, and Jiaya Jia. Generalized few-shot semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11563–11572, 2022. 1, 2, 4, 5, 6

[5] Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia. Prior guided feature enrichment network for few-shot segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(2):1050–1065, 2020. 4

[6] Gengwei Zhang, Guoliang Kang, Yi Yang, and Yunchao Wei. Few-shot segmentation via cycle-consistent transformer. *Advances in Neural Information Processing Systems*, 34:21984–21996, 2021. 1

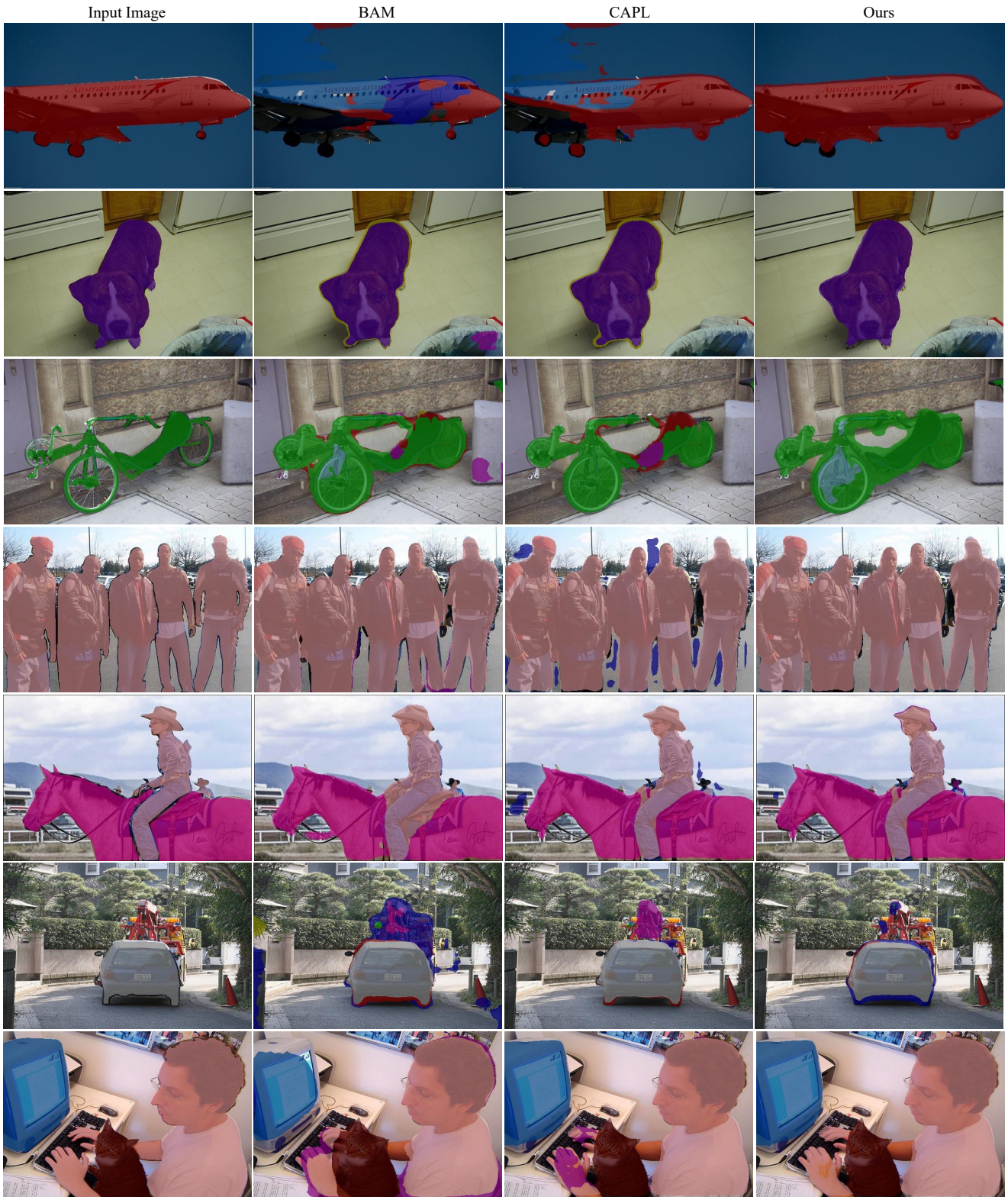


Figure 3. Qualitative results of BAM [2], CAPL [4], and our method on PASCAL-5^i benchmark with large object appearance variations. Zoom in for details.

Figure 4. Qualitative results of BAM [2], CAPL [4], and our method on COCO-20^i benchmark with large object appearance variations. Zoom in for details.