(R1) Robustness analysis for supervised pre-training, self-supervised pre-training, and self-training. We agree that robustness for pre-training and self-training is an interesting direction. However, unlike ImageNet, there is no robustness benchmark for detection and segmentation. This can be very interesting future follow-up work.

(R1, R4) The performance in Table 3 and Table 4 does not match. The backbone models are not the same in Table 3 (EfficientNet-B7) and Table 4 (ResNet-50), because SimCLR does not have an EfficientNet checkpoint. We will revise and make it more clear in the caption.

(R1) Are the images in Fig. 5 all used as the pseudo-labeled images for the student training? Yes, that’s correct. We do not filter any images even if they only contain the background class.

(R2, R3, R4) Analysis of why self-training performs better than pre-training. Our discussion section attempted to address our two hypotheses on why self-training outperforms pre-training. We will update the section to more clearly address the question. Hypothesis 1: Joint optimization of human and pseudo labeled data; 2: Task alignment.

Joint Optimization (1): In Table 7 we study the benefits of joint optimization with two experiments: jointly training ImageNet classification and COCO object detection and pre-training with ImageNet classification and fine-tuning on COCO. Our results reveal that jointly training the ImageNet objective with COCO is more effective than pretraining a model with the ImageNet objective and then fine-tuning on COCO.

Task Alignment (2): Table 8 studies the importance of task alignment. Pascal consists of two parts: a standard train set ("train") and a train set labeled with a different distribution of human annotations ("trainaug"). We observe training a teacher on "train" then relabeling "trainaug" outperforms using the original annotations of "trainaug" when training on both datasets concurrently (84.8 vs. 86.7 mIoU). We observe using targeted pseudo labels is more useful than using ground truth human labels that do not well match the target labels.

(R2) Which is more important: strong data augmentation or self-training? (1) Self-training is quite additive to data augmentation and (2) self-training is more general than data augmentation methods as it does not require domain knowledge. In Table 2 the best data augmentation yields 5.1AP improvement and self-training over a +1.3 AP improvement across all augmentation methods. Therefore we do not see a tradeoff between the two methods and suggest using both data augmentation and self-training. Furthermore, self-training can also be done without any dataset knowledge, where data augmentation methods have to be crafted according to the task at hand. If we apply self-training to a self-driving car dataset, the data augmentation method needs to change whereas self-training can stay the same.

(R2) Will instance segmentation on COCO show similar conclusions as box detection and semantic segmentation? Our hypothesis is yes. Unfortunately, we do not have the experiments to answer the question.

(R2, R4) Will the conclusions be similar to fine-grained recognition problem or open-set embedding learning problem, e.g., face recognition? Great question! We want to argue that it is possible to apply self-training to open-set recognition (such as face recognition), where the training labels are typical defined by similar/dissimilar pairs of examples. The teacher model can also label similar and dissimilar examples in the unlabelled dataset for open-set tasks.

For fine-grained/open-set recognition problem, there can be more noise in the pseudo labels. We observe self-training can work well even when the pseudo labels are noisy. For example, Figure 5 shows several wrong pseudo segmentation labels (e.g., mislabel a saw with the bird) because the concept does not exist in the teacher model. Nevertheless, Table 13 shows improvements using these examples for self-training. This hints that self-training has a certain degree of robustness against noise in pseudo labels caused from domain or label space shift.

(R2) Does self-training benefit low capacity models? We experiment with a wide variety of different model capacities in our paper: ResNet-50, ResNet-101, EfficientNet-B7, and EfficientNet-L2. All these models show consistent benefits when applying self-training. However, we do not experiment with mobile size models such as MobileNet. In the classification domain [12] applied self-training to a mobile sized model and sees good improvements.

(R3) Additional experiments to control the domain shift, swap the labeled and unlabeled datasets, and use SimCLR checkpoint for self-training. Thanks for suggesting extra experiments. Due to the limitation of time and page limit, we are not able to try all these ideas. Chen et al. [a] shows promising results combining self-supervised training and self-training.

(R3) Comparison of DeepLab and our model. Here we want to show that self-training can work for semantic segmentation in a high performance regime by using the EfficientNet + FPN architecture. Therefore having a baseline EfficientNet with and without self-training proves our point.