We thank the reviewers for valuable feedback. Before addressing individual comments, we clarify common concerns.

Evaluation Protocol: The most ambitious aim of self-supervised learning is to create universal visual representations that do not require end-to-end fine-tuning, but are instead portable to new tasks merely through additional of a shallow auxiliary network. Moreover, "image-level" vs "pixel-level" training has no bearing on the validity of evaluating with respect to this stricter protocol. Any method that uses a CNN learns more than just "image-level" representations; for example, a CNN's internal activations can always be converted into per-pixel embeddings using hypercolumns (as we do; see L189). While we agree that adding more fine-tuning results is important for expanding comparison to past work, and will do so, we believe our more ambitious evaluation methodology is fundamental to driving progress in the field.

Dataset Properties: Progress in supervised object segmentation has been driven by a series of datasets with increasing label complexity per scene, from PASCAL to COCO to LVIS[1]. It is reasonable to suspect a similar trend will hold for self-supervised learning, in which case it will be essential to move away from methods such as [19,5,7], which build pipelines dependent upon the single-object bias of the dataset (ImageNet). Corroborating this view are recent results[2] suggesting even supervised ImageNet pretraining is of minimal value for learning visual tasks on complex scenes.

Experiments: As we are interested in learned embeddings that *universally* and *directly* fit a broad range of tasks (*e.g.* instance tracking, segment search, semantic segmentation), we freeze the backbone in most experiments. We have now also run the end-to-end fine-tuning experiment on COCO+VOC pretrained backbone, following the design in [19] to replace DeepLabV3 head with two stacked convolution layers. Results are: ours 47.2 vs MoCo 46.9 mIOU. We use a simpler siamese training architecture, instead of a momentum encoder and memory bank, yet achieve comparable results to MoCo. With frozen pretrained backbone, we also run instance segmentation tasks on COCO2014, following Mask RCNN in [19]. Results of ours vs MoCo: box AP 18.62 vs 14.75 and mask AP 17.82 vs 14.31.

Suggested by **R4**, we retrain our model on COCO+VOC with HED edges and achieve 49.9 mIOU in above mentioned end-to-end fine-tuning settings and 48.8 (raised from 46.5) mIOU in the frozen backbone experiment. Suggested by **R3**, to demonstrate the efficiency of proposed pixel-wise optimization, we train MoCo on VOC only for the same number of iterations (32K) and get 26.7 mIOU (ours is 43.5 mIOU) evaluated by freezing the backbone.

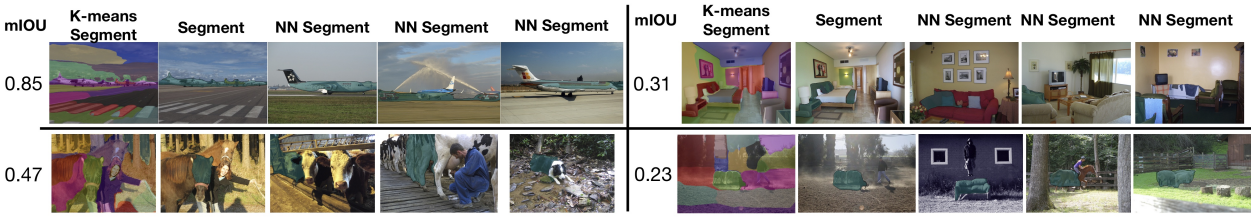

Figure 1: Visualization of success and failure cases denoted by mIOU on segment search task.

R1 – COCO and VOC less curated? LVIS[1] analyzes statistics of COCO images, finding many instances per-image and a long-tailed category distribution, which supports our characterization. While Instagram-1B does not appear to be publicly available, we worry that Instagram images might suffer from significant photographer bias.

R2, R3, R4 – SegSort. SegSort focuses on a different task: using contours to refine ImageNet pre-trained features, producing semantic segmentation. Our task is to learn pixel-wise semantic-aware embeddings from scratch. To compare, we evaluate SegSort with its backbone initialized from scratch. Performance in "semantic segment retrieval", the truly unsupervised equivalent of SegSort's "unsupervised semantic segmentation" is this lower than their reported scores.

R1, R2 – Partial ImageNet results. We will update the final version to reflect the full 200 training epochs.

R1, R2 – Sampling details. We first sample regions, then a fixed number of pixels within chosen regions. Probability should be summed over regions, *i.e.* $\sum_{R_b} P(j \in Pos(i)) = 1$. Pixels could be sampled as positive and negative at the same time. Instead of a memory bank, we compute the representation from both original and augmented images.

R1 – Distance in tree. We compute $d_T(R_i, R_j)$ following Sec. 4.2 of [2], originally introduced in Sec. 2-4 of [1].

R3 – Hyperparameters. We pick σ_p using limited search (0.2 - 1.0, spaced by 0.2). For MoCo training on COCO + VOC, we also ran a limited hyperparameter search and did not see significant impact on results.

Citations. R1: We will add suggested papers. **R3:** Missed citation is indeed Li *et al.*, CVPR 2016.

Misc. R1: Table 1 row 9 should be 36.15. We will update Fig. 3, 4, 5 with best/worst cases (for Fig. 4, some examples are shown above in Fig. 1). **R2:** 'Image' under 'Cross-Image Negative Sample' refers to treating the image-based embedding as negative features. 'spatially-extended' means we preserve the spatial dimension in the embedding and output a feature map. **R3:** We fine-tune on PASCAL $train\_aug$ and evaluate on PASCAL val.

[1] A. Gupta, P. Dollar, R. Girshick. LVIS: A Dataset for Large Vocabulary Instance Segmentation. CVPR, 2019.

[2] K. He, R. Girshick, P. Dollar. Rethinking ImageNet Pre-training. ICCV, 2019.