Supplementary Materials - Just Label What You Need: Fine-Grained Active Selection for P&P through Partially Labeled Scenes

Sean Segal[12*], **Nishanth Kumar**[3*], **Sergio Casas**[12], **Wenyuan Zeng**[12], **Mengye Ren**[12], **Jingkang Wang**[12], **Raquel Urtasun**[12]
[†] Waabi[1], University of Toronto[2], MIT CSAIL[3]
{ssegal, scasas, wzeng, mren, jwang, urtasun}@waabi.ai, njk@csail.mit.edu

Abstract: In these supplementary materials, we provide additional details for training P&P models from partial supervision. We also present more experimental details and results, including an additional larger-scale experiment, selection statistics, and many qualitative examples of labels selected by fine-grained active selection. In our video **supplementary.mp4**, we provide a narrated overview of our method and provide qualitative examples of labels selected by active selection.

1 Additional Details for Partial Supervision Loss

As described in Equation (6) of the paper, we modify the standard P&P loss to support training from partial supervision. The final loss is the sum of a loss, l_P applied to each positive example (i.e. actor label), and a background loss l_B which discourages the network from outputting detections on negative regions inside the labeled region, R. Next, we describe the implementation of these losses in detail.

Positive Loss: In our implementation, the loss applied to all positive examples includes a probabilistic negative loss likelihood loss on the predicted Gaussian mixture (see [12] for details), a smooth L1 regression loss on the bounding box parameters, and a cross entropy loss for the classification detection task,

$$l_P(\mathbf{y}_i, \hat{\mathbf{y}}_i) = l_{\text{NLL}}(\mathbf{y}_i, \hat{\mathbf{y}}_i) + l_{\text{reg}}(\mathbf{y}_i, \hat{\mathbf{y}}_i) + l_{\text{cel}}(\mathbf{y}_i, \hat{\mathbf{y}}_i) \ . \tag{1}$$

Background Loss: Following [21], we employ hard negative mining to train the classification component of the detector. Importantly, we only consider negative anchors **within** the labeled region R, since regions outside of R may contain positive examples which have not yet been labeled. Specifically, we first remove any positive or ignore anchors, then we additionally filter any anchors that are not contained within the labeled region R. Finally, we take up to $3\times$ as many negative examples as there are positives and select those with the highest classification score (i.e., the hardest examples).

Loss Aggregation: When aggregating the loss over a batch, one common approach (e.g., used in [21]), is to first average over all actors in the same scene, then take the average over scenes in the batch. This works well when training on fully labeled scenes, but can be problematic in the partially labeled setting, since there can be large imbalances between the number of labeled actors in each scene. For example, without any constraints during selection, a partially labeled dataset might have some scenes with only a single actor labeled and others fully labeled. The averaging

[*] Equal Contribution
[†] This work was done by all authors while at Uber ATG

5th Conference on Robot Learning (CoRL 2021), London, UK.

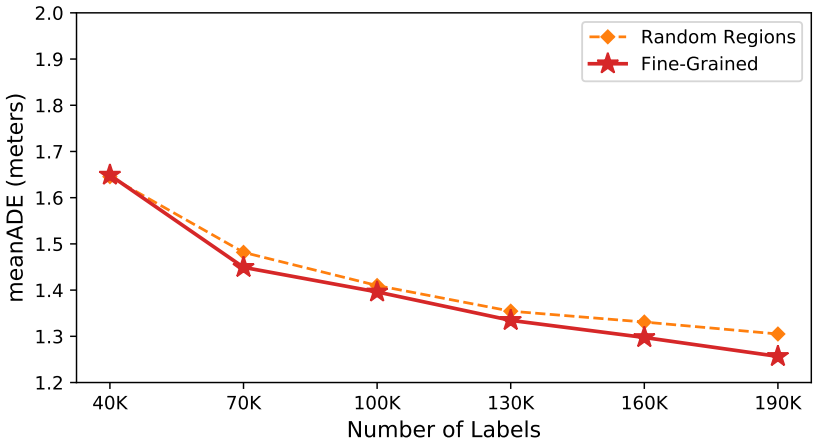

Figure 1: **Additional Large-Scale Run:** Prediction performance of random regions vs. fine-grained selection over N = 5 active learning iterations. 30K labels are selected at each iteration.

approach described above will upweight actors in scenes with less labels, which is undesirable. As a solution, we propose to weight all actors equivalently in the loss by reducing the loss with a sum instead of average. Under this approach, if a randomly sampled mini-batch has more dense supervision, it will contribute more to the loss. While we find that this significantly improves the performance of models trained from partial labeling, we note that we still find it challenging to train P&P models from datasets which have large imbalances in supervision from scene to scene, which was an additional motivation for the sparsity constraint which we ablate in Figure 6 of the paper.

2 Experimental Details

2.1 Baseline Details

Learning Loss Baseline: Following [1], we train an additional loss prediction module to learn to predict the loss of unlabeled examples. We follow the network design of the original paper, but adapt it to take the intermediate features from the backbone of our P&P model. We train the module to predict only the prediction loss (not perception related losses) and re-tune hyperparameters, resulting in a margin $\xi = 1.0$ and loss prediction weight $\lambda = 0.001$. For scoring, we run inference on all unlabeled examples and select those with the highest predicted loss.

CoreSet Baseline: We additionally compare against the common diversity-based approach of `Core-Set` [21] selection. Rather than score examples independently, the method seeks to select a representative sample based on distances between examples in a learned space. To compute distances, we leverage the learned feature representations of the network to compute distances between examples. For efficiency, we leverage the k-Greedy center variant of the algorithm from the paper, which we find is most commonly used in practice.

2.2 Planning Metrics

Metric Definitions: In Table 1 of the main paper, we calculate 5 common planning metrics to evaluate the downstream impact of active selection on planning performance. First, we measure the **collision** rate which captures the percentage of SDV's plans which collide with an obstacle. To capture similarities to human driving, we also compute the **L2** distance between the human driven trajectory and the SDV's plan. To capture the comfort of the planned trajectory, we measure the **lateral acceleration** and **jerk** of the SDV's planned trajectories. Finally, we measure the SDV's **progress** along the route. All metrics are computed over a 5 second planning horizon.

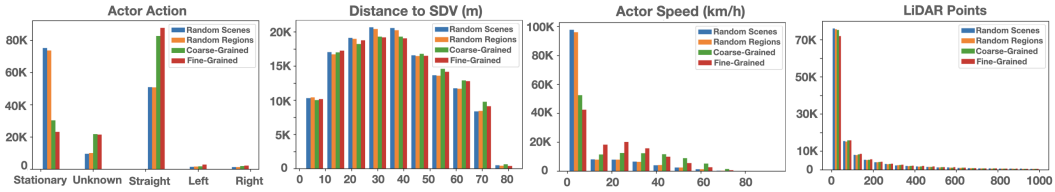

Figure 2: **Selection Statistics:** Statistics of the labels selected by each active selection approach at the final active learning iteration.

3 Additional Results

Additional Active Learning Run: In this experiment, we compare fine-grained active selection to the most competitive baseline (i.e. random regions) starting from a different, randomly initialized labeled pool than the one used in the main paper's results. To test generalization at a larger-scale, we select $B = 30$K labels per active learning iteration (instead of the 20K selected at each iteration in the main paper). Figure 1 shows the results in the same format at Figure 2 (right) in the main paper. We find our results are consistent with those presented in the main paper: fine-grained selection consistently outperforms random regions and the gap becomes more apparent at later active learning iterations.

Selection Statistics: Figure 2 contains histograms of label statistics selected by each method after the final iteration of active learning. `Core-Set` and `LearnLoss` are omitted due to similarities with `Coarse-Grained`. We compute the histograms based on label metadata, including their high level action (driving straight, left, right, stationary), the vehicle speed, distance to the SDV, and the number of LiDAR points contained inside the bounding box. As expected, we notice that active-selection methods tend to sample more non-stationary vehicles and vehicles further from the SDV. This effect is more apparent for fine-grained selection methods due to the additional flexibility provided by the partially labeled setup. One potential downside of fine-grained selection is that it will be biased towards regions with actors detected by the current model. While this leads to sampling more visible labels (i.e., labels with more LiDAR points), we do not see this affect model performance.

Qualitative Examples: Below, we provide additional qualitative examples of labels selected by fine-grained active selection, in the same format as Figure 5 of the main paper. Examples were selected at random from the final set of selected labels. We notice that fine-grained active selection primarily selects moving actors and tends to select vehicles with interesting behaviors.