Motion Inspired Unsupervised Perception and Prediction in Autonomous Driving: Supplementary Materials

Mahyar Najibi*, Jingwei Ji*, Yin Zhou*†, Charles R. Qi, Xinchen Yan, Scott Ettinger, and Dragomir Anguelov

Waymo LLC
{najibi,jingweij,yinzhou,rqi,xcyan,settinger,dragomir}@waymo.com

1 Implementation Details of Auto Meta Labeling

The Auto Meta Labeling pipeline has the following components: object proposal by clustering, multi-object tracking and amodal box refinement based on shape registration. In the object proposal step, we use DBSCAN for both clustering by point locations and by scene flows. Both clustering methods use Euclidean distance as the distance metric. The neighborhood thresholds ϵ_p and ϵ_f are set to be 1.0 and 0.1, respectively. The minimum flow magnitude $|\mathbf{f}|_{min}$ is set to 1m/s, so as to include meaningful motions without introducing too much background noise. Our tracker follows the implementation as in [3]. We use bird's eye view (BEV) boxes for data association and use Hungarian matching with an IoU threshold of 0.1. In shape registration, we use a constrained ICP [1] which limits the rotation to be only around the z-axis. We have compared the effect of contrained and unconstrained ICP in AML ablation study. The search grid for translation initialization is decided by the target box dimensions on the xy-plane, *i.e.* the length $l_{\mathbf{b}_{tgt}}$ and the width $w_{\mathbf{b}_{tgt}}$ of the target bounding box. We enumerate translation initialization $\mathbf{T}_j$ in a 5×5 grid covering the target bounding box region with a list $\mathcal{T}_x$ of strides as $[-l_{\mathbf{b}_{tgt}}/2, -l_{\mathbf{b}_{tgt}}/4, 0, l_{\mathbf{b}_{tgt}}/4, l_{\mathbf{b}_{tgt}}/2]$ and a list $\mathcal{T}_y$ of strides $[-w_{\mathbf{b}_{tgt}}/2, -w_{\mathbf{b}_{tgt}}/4, 0, w_{\mathbf{b}_{tgt}}/4, w_{\mathbf{b}_{tgt}}/2]$. Each computation of ICP outputs an error ϵ_j, which is defined as the mean of the Euclidean distances among matched points between the source and the target point sets.

2 Ablation Study on Unsupervised Flow Estimation

In this section we provide additional ablation studies focusing on our unsupervised flow estimation method, NSFP++.

Static point removal As mentioned in Section 3.1 of the main paper, we apply static point removal prior to unsupervised flow estimation. This step is

* Equal contribution
† Corresponding author

Table A1. Ablation study on different components in the proposed local flow estimation. BQ stands for the proposed box query strategy, which contains two steps, the first being expansion and the second being pruning. Local consistency represents the local consistency loss among flow predictions within each point cluster.

Variants of NSFP++			EPE3D ↓	θ (rad) ↓	mIoU ↑
BQ w. Expansion	BQ w. Pruning	Local Consistency			
			0.020	0.515	0.404
✓			0.023	0.560	0.552
✓	✓		0.018	0.504	0.571
✓	✓	✓	0.017	0.474	0.586

Table A2. Flow comparison with the fully supervised model.

Method	EPE3D (m) ↓	θ (rad) ↓	mIoU ↑
Fully Supervised Network	0.005	0.062	0.826
Unsupervised NSFP++ (ours)	0.017	0.474	0.586

designed to achieve a high precision to avoid removing dynamic points in the early stages of our pipeline. Here, we compute the precision/recall of this step on the WOD [4] validation set. We define ground-truth dynamic/static labels based on the available ground-truth bounding boxes [2]. Dynamic points are defined as those with a ground-truth flow magnitude larger than $|\mathbf{f}|_{min}$, and the remaining points belonging to any ground-truth box are assigned to the static class. Our static point removal step has a precision of 97.2%, and a recall of 62.2%, validating the high precision of this step in determining the static points.

Local flow estimation We also conduct ablation study to validate the effectiveness of the proposed components in the local flow estimation step, *i.e.,* box query with expansion followed by pruning and local consistency loss. As illustrated in Table A1, box query with expansion (second row) effectively boosts mIoU from 0.404 to 0.552 but suffers from higher 3D end-point error (EPE3D) and mean angle error (θ), compared to the method without using box query (first row). This is due to the fact that the expanded query region can capture more matching points but at the cost of including irrelevant points. With the proposed pruning scheme (third row), all metrics are significantly improved compared to the previous two rows. Finally, by adding local consistency loss (fourth row), we obtain the best performance across the board.

Comparison with the fully supervised model In this subsection, we compare our unsupervised flow estimation method with the fully supervised scene flow model used in Section 4 of the main paper. Table A2 shows the comparison. As expected, the supervised model outperforms our unsupervised NSFP++

Table A3. Comparisons of different variants of components in the AML pipeline. All methods are evaluated on the WOD validation set.

AML Variants	3D mAP		2D mAP	
	L1	L2	L1	L2
Filtered by flow + Clustering by position	25.5	24.6	32.4	31.2
Spatio-temporal clustering	**30.4**	**29.2**	**36.7**	**35.3**
Regis. w/o init.	32.2	31.0	36.6	35.3
Regis. w/ R init. by flow heading	33.2	31.9	37.4	36.0
Regis. w/ T init. by grid search	35.2	33.8	39.3	37.9
Regis. w/ Unconstrained ICP	34.3	33.0	38.5	37.1
Regis. w/ RT init. & constrained ICP [1] (**Full AML**)	**36.9**	**35.5**	**40.5**	**39.0**

method which does not use any human annotations. However, as shown in Table 2 of the main paper, the AML pipeline can robustly use our unsupervised NSFP++ predictions and eventually achieves comparable results to the counterpart using a supervised flow model on downstream tasks (e.g., L1 mAP of 42.1 for unsupervised *v.s.* 49.9 for supervised in the object detection task).

3 Ablation Study on Auto Meta Labeling

To examine the design choices in the AML pipeline, we compute the detection metrics on the auto labels generated by our full AML pipeline and several baselines (Table A3). Note that the numbers reported in Table A3 are from evaluation on auto labels, rather than on the predictions by trained detectors. *Filtered by flow + Clustering by position* is a baseline where we generate auto labels only using this clustering method. Compared to our spatial-temporal clustering method described in Algorithm 1 in the main paper, this baseline does not perform clustering on the estimated flows and as a result it leads to under-segmentation and lower performance.

We also carry out experiments on variants of shape registration. *Regis. w/o init.* is a baseline where we have no initialization when performing constrained ICP. Adding either rotation initialization by flow heading (*Regis. w/ R init. by flow heading*) or translation initialization by grid search (*Regis. w/ T init. by grid search*) improves the quality of auto labels. Another baseline, *Regis. w/ Unconstrained ICP*, is applying both $\mathbf{R}$ and $\mathbf{T}$ initializations but uses an unconstrained ICP such that 3D rotations are allowed when aligning the source and the target point sets. We find that limiting the rotation to be only around z-axis generates auto labels with a higher quality. Finally, our full AML (*Regis. w/ RT init. & constrained ICP*) outperforms all other variants. Compared to the 3D detection results in the main paper (see Table 2 in the main paper), we find that the object detector achieves higher mAP than the auto labels it is trained on. The reason is that auto labels by design pursue high recall while contain some false positives in the background due to inaccurate flow or noise in the environment. As these false positive labels do not form a consistent data pattern, the object detector learns to focus only on auto labels with common

patterns, such as vehicles and VRUs, and assign high confidence scores to these objects at inference time.

4 Qualitative Analysis

4.1 Auto Meta Labeling and Unsupervised Object Detection

Fig. A1 shows four examples from the WOD validation set comparing ground truth, auto labels and unsupervised object detection results. In our unsupervised setting, both the auto labels and object detectors localize objects in a class-agnostic manner and are not limited by certain categories. In example (a) we show that auto labels and object detectors capture both pedestrians and vehicles.

In example (b), we demonstrate that even though there is *false positive* non-zero flow estimation, in AML we filter out many of these clusters during tracking and post-processing where very short tracks are dropped. The resulting detector has also learned to ignore clusters of false positive flows. This example also shows that both auto labels and object detectors can infer the amodal boxes of some objects with only partial views.

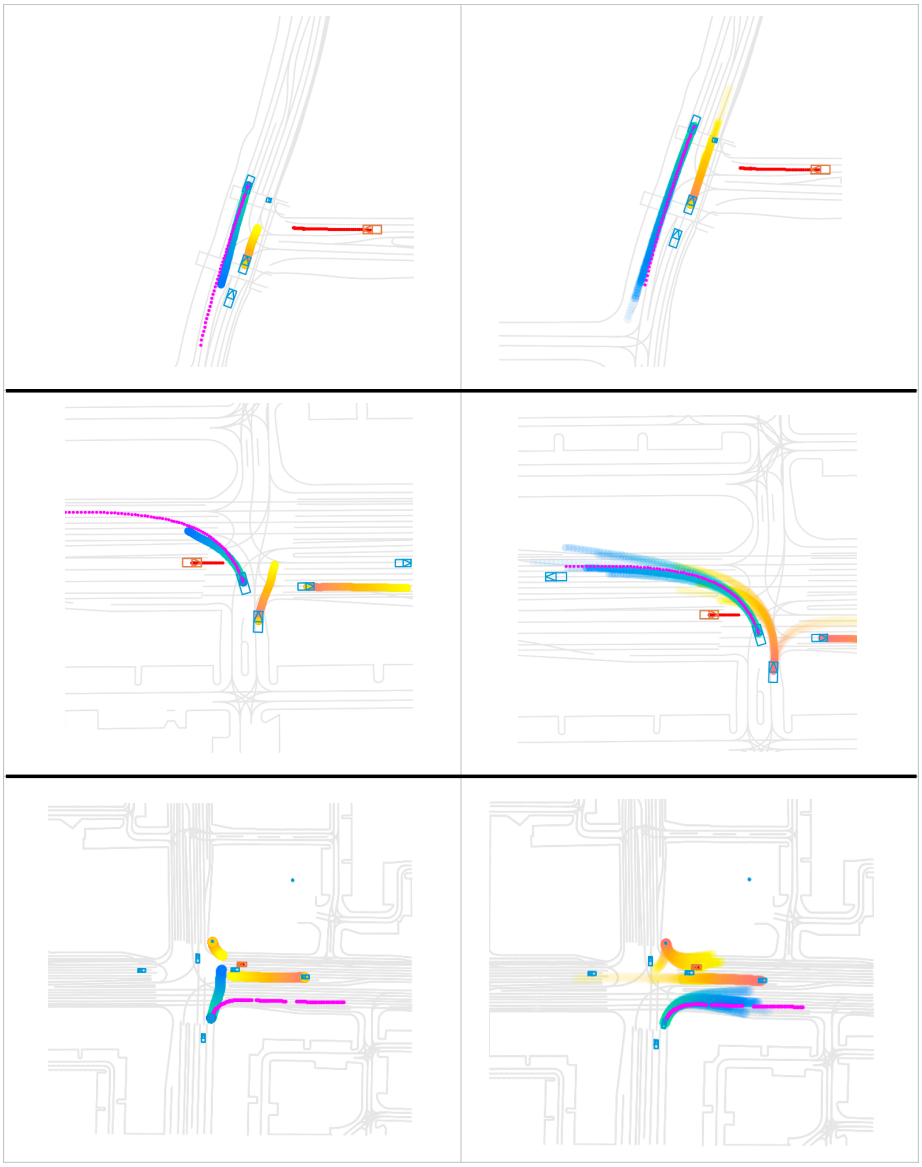
Sometimes the unsupervised flow estimation captures *true positive* motion on points that are beyond the predefined categories in the ground truth. In example (c), a pedestrian is walking with a stroller while *stroller* is not a class included in the ground truth labels and therefore no bounding box is annotated around the stroller. NSFP++ has estimated the flow on the stroller, enabling AML and detectors to localize it. Since the stroller is held by the pedestrian with a similar speed, the clustering by design does not separate them apart. Clearly, it is safety-critical for autonomous vehicles to understand such moving objects in the open-set environment.

Example (d) shows a failure case where the detector could not confidently detect a cyclist. Although the auto labels have captured it, cyclists are less common than pedestrians and vehicles in the training set, which leads to inferior performance. We encourage future work to tackle the data imbalance issue under the unsupervised setting. Another failure pattern is that bounding boxes in auto labels tend to be larger than the actual size, due to the fact that temporal aggregation can include noise points. More advanced shape registration methods may help reduce noise and we leave it for future work.

4.2 Open-set Trajectory Prediction

Fig. A2 and A3 show behavior prediction qualitative results on the WOD validation set. For each example scenario, we show the trajectory predictions of two models, *i.e.,* one trained only with a human-labeled category (the first column) and the other one trained with the combination of available human-labels and our AML auto labels for all other moving objects (the second column). The red and magenta trajectories represent the ground-truth routes taken by the autonomous vehicle and by an agent of interest, respectively. The blue and yellow trajectories are the possible predictions for the agent of interest and other

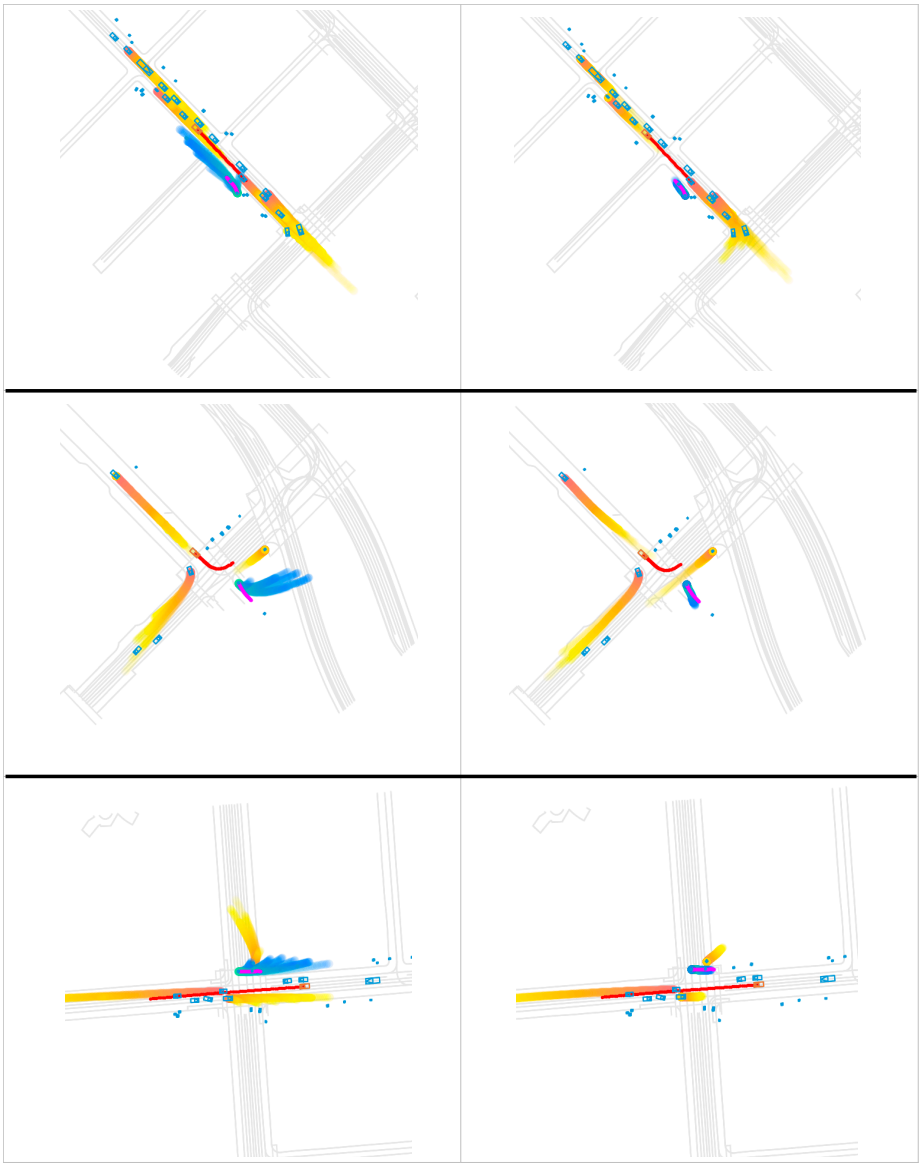
Fig. A1. Visualization of auto labels and detection predictions compared with the ground truth of moving objects. Points are colored by flow magnitudes and directions. Dark points are static. (a) The class-agnostic auto labels and unsupervised object detectors capture objects of multiple categories. (b) Although false positive flows occur, AML filters out many of them if they are inconsistent, and the detector learns to ignore these false positive flows. (c) Although the ground truth does not cover categories beyond vehicle, pedestrian, and cyclist, auto labels and our detector can capture open-set moving objects, such as the stroller. (d) An failure case that the detector may not be confident on objects with limited data amount, such as cyclists.



Trained with human labeled VRU

Trained with AML (ours) + human labeled VRU

Fig. A2. Behavior prediction qualitative analysis. Trajectory predictions on three example scenarios for a model trained with human labeled VRUs *v.s.* a model trained with a combination of human labeled VRUs and our generated autolabels. Red and magenta dotted trajectories represent the ground-truth routes of the autonomous vehicle and agents, respectively. Blue and yellow trajectories are the predictions for the agent of interest and other agents, respectively.



Trained with human labeled vehicles

Trained with AML (ours) + human labeled vehicles

Fig. A3. Behavior prediction qualitative analysis. Trajectory predictions on three example scenarios for a model trained with human labeled vehicles *v.s.* a model trained with a combination of human labeled vehicles and our generated autolabels.

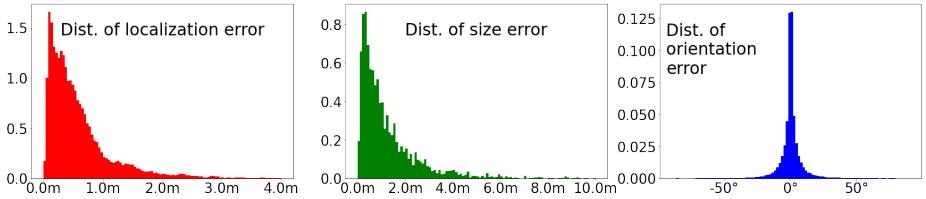


Fig. A4. Error distributions. y axis is probability density.

agents in the scene. Fig. A2 shows three examples where human labels are available for the VRU category. As can be seen in all three examples, without using our unsupervised auto labels, the model tends to erroneously underestimate the speed (*e.g.* the first row), have difficulty in predicting trajectories consistent with the underlying roadgraph (*e.g.* the second row), and generating dangerous pedestrian-like trajectories along the pedestrian crosswalk (*e.g.* the third row). Fig. A3 shows the results when human labels are available only for the vehicle category. Similarly, when the model is only trained on the human labels (the first column), it cannot generalize well to the VRU class, predicting fast speeds and vehicle-like trajectories for VRUs. However, in both scenarios, adding auto labels (the second columns in Fig. A2 and A3) satisfactorily overcomes these errors, showing the effectiveness of our auto labels for training behavior prediction models in the open-set environment.

5 Failure Analysis

In this section, we analyze the factors causing failure cases. Under threshold IoU=0.4, the precision/recall of our auto meta labels is 0.69/0.50. Part of the failure cases come from (1) false positive predictions that do not match any ground truth boxes; (2) false negatives where ground truth boxes are entirely missed. Moreover, there are predicted boxes overlapping with ground truth boxes while their IoUs are lower than the threshold. To have a better understanding, we breakdown 3D bounding box dimensions into three groups: localization (box center x, y, z), size (box length l, width w, height h), and orientation (BEV box heading r). Then, we summarize the distributions of localization, size, and orientation errors of the generated bounding boxes which overlap with at least one ground truth box (Fig. A4). The errors are computed between each pair of a predicted box and the ground truth box that has the highest IoU with the predicted box.

Localization. The localization error is defined as

$$\epsilon_{localization} = \sqrt{(x_{pr} - x_{gt})^2 + (y_{pr} - y_{gt})^2 + (z_{pr} - z_{gt})^2}. \quad (1)$$

As shown in Fig. A4, most of the localization errors are within 1.0 meter.

Table A4. Comparison between an oracle with GT box coordinates and baselines switching localization/size/orientation coordinates into AML predictions in turn. The performance drops show that the localization and size errors are dominant.

	3D mAPH@IoU=0.4
(Oracle) GT localization + GT size + GT orientation	46.1
Predicted localization + GT size + GT orientation	39.7 (-6.4)
GT localization + Predicted size + GT orientation	39.7 (-6.4)
GT localization + GT size + Predicted orientation	44.5 (-1.6)

Size. The size error is defined as

$$\epsilon_{size} = \max\{|l_{pr} - l_{gt}| + |w_{pr} - w_{gt}| + |h_{pr} - h_{gt}|\}. \tag{2}$$

Many predictions have relatively high size errors. This is often caused by inclusion of noisy points in the registration step or missing parts of an object if the parts are always invisible throughout the object track.

Orientation. The orientation error is defined as

$$\epsilon_{orientation} = r_{pr} - r_{gt} \tag{3}$$

The orientation errors are generally small, as the orientation of each object is determined by the direction of the scene flows averaged over all points within the object bounding box. This error distribution verifies the quality of the unsupervised scene flows.

To find out the dominant factors leading to wrong auto meta labels, we construct several baselines by modifying the predictions and measure their label quality. The baselines are as follows:

1. (Oracle) GT localization + GT size + GT orientation: we replace the 7D values (x, y, z, l, w, h, r) of each predicted box with the values of its best matched ground truth box if any;
2. Predicted localization + GT size + GT orientation: we replace the (l, w, h, r) of each predicted box with the ground truth values. Comparison with the oracle will show the impact of localization errors;
3. GT localization + Predicted size + GT orientation: we replace the (x, y, z, r) of each predicted box with the ground truth values. Comparison with the oracle will show the impact of size errors;
4. GT localization + GT size + Predicted orientation: we replace the (x, y, z, l, w, h) of each predicted box with the ground truth values. Comparison with the oracle will show the impact of orientation errors.

We report the 3D mAPH@IoU=0.4 on the above baselines as mAPH additionally reflect the quality of heading prediction. We found that localization and size errors are dominant factors and future work may focus on improving the quality of auto labels on these fronts.

References

1. Groß, J., Ošep, A., Leibe, B.: Alignnet-3d: Fast point cloud registration of partially observed objects. In: 3DV (2019) 1, 3
2. Jund, P., Sweeney, C., Abdo, N., Chen, Z., Shlens, J.: Scalable scene flow from point clouds in the real world. IEEE Robotics and Automation Letters **7**(2), 1589–1596 (2022). https://doi.org/10.1109/LRA.2021.3139542 2
3. Qi, C.R., Zhou, Y., Najibi, M., Sun, P., Vo, K., Deng, B., Anguelov, D.: Offboard 3d object detection from point cloud sequences. In: CVPR (2021) 1
4. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., Anguelov, D.: Scalability in perception for autonomous driving: Waymo open dataset. In: CVPR (2020) 2