Supplementary Material for Uncertainty-Guided Never-Ending Learning to Drive

Abstract

*This supplementary document focuses on **more detailed analysis** regarding the methodology and experiments presented in the main paper, including **implementation details** regarding network architecture and training (Sec. 1) and discussion of **supportive quantitative and qualitative analysis** (Sec. 2). Please see our **supplementary video** for additional motion planning policy results. To facilitate future research in scalable, generalized, and adaptive planning policies, all of our code, models, and datasets are available at https://infdriver.github.io/.*

1. Implementation Details

In this section, we provide additional details regarding the architecture of our inverse dynamics and motion planning models (Sec. 1.1), the incremental training protocol (Sec. 1.2), the baseline implementation (Sec. 1.3), the training dataset leveraged throughout our experiments (Sec. 1.4), and the transformation of converting ego-vehicle pose into waypoints, speed, and command (Sec. 1.5).

1.1. Network Architecture

Inverse Dynamics Model Architecture: We structure self-training a never-ending policy as two-stage learning. First, a diverse ensemble of inverse dynamics, i.e., visual odometry (VO), models is used for pseudo-labeling the incoming image stream. Subsequently, a motion planning policy model is trained with the pseudo-labels as targets. This decomposition enables ∞-Driver to effectively and directly learn from demonstrated decisions in the frames, i.e., as opposed to pre-training-based strategies which do not integrate decision-making behavior [20]. Moreover, an inverse dynamics model can be used to **infer speed and conditional command which are required as input** for state-of-the-art goal-oriented motion planning policies (e.g., [7–10]). By leveraging image-only data to obtain such inputs, our method is particularly suitable to large-scale settings where GPS-based data may be noisy or intermittent. The network architecture of inverse dynamics model is inspired by the efficient DeepVO [19] architecture, which proposes a direct regression, learning-based model without relying on any known camera parameters. We extend the model to include a probabilistic model over the rotation predictions [14]. We find it sufficient to only account for uncertainty in rotation, and probabilistic modeling of translation predictions, e.g., for removing or re-labeling noisy samples, did not provide further gains. Moreover, based on our experiments, we find our learning-based architecture to significantly outperform other VO baselines in our settings, i.e., over ORB-SLAM [6] and DROID-SLAM [17]. Specifically, we leverage a **FlowNet as the encoder and a branched decoder**. One decoder predicts the Matrix Fisher distribution and comprises Multi-layer Perceptron (MLP) with three stacked fully-connected (FC) layers with hidden dimensions 128, 128, and nine, respectively, each with tanh activations. The second decoder is used to predict the rotation and translation, comprising of two stacked LSTM layers each with 128 hidden dimension and a final FC layer with a six-dimensional output (parameterizing 3D rotation and translation). To obtain robust pseudo-labels and estimate epistemic uncertainty, we train and incrementally update a set of $M = 5$ models with identical architecture but initialized based on different random seeds. A discussion on how we efficiently promote diversity among the models in training can be found in Sec. 1.2.

Planning Policy Model Architecture: While our generalized framework can support any policy model, we design our policy model similarly to prior conditional image-based driving policies [8, 21]. Given image, speed, and driving command, the image is first encoded with a convolutional ResNet-34 backbone [8, 12] outputting a 512 by 8 by 13 descriptor which is then

concatenated with the current speed. Following the concatenation, we implement a standard conditional prediction head (as in Chen et al. [8]). We use three commands, where each head comprises three deconvolution layers and a spatial softmax over predicted waypoint heatmaps to generate Bird's Eye View (BEV) waypoint coordinates based on the driving command. We note that we regress BEV waypoints directly instead of relying on a known homographic transformation. We refer the reader to Chen et al. [8] for additional details.

1.2. Training Protocol

We first train our ensemble of inverse dynamics models on nuScenes-Boston, and subsequently incrementally update over incoming images (e.g., videos from YouTube). For the inverse dynamics model training, the batch size is set to 16 and the learning rate to 0.0005, optimized via SGD for 15 epochs. For the motion planning model training, the batch size is set to 96 and the learning rate to 0.001, optimized via Adam for 30 epochs. The size of $\mathcal{S}$ is set as 500,000. $f^{\text{inv}}_{\psi_m}$ incrementally trains on ten million YouTube images (extracted from videos at 10Hz) from various weather conditions and driving scenarios. The time window size in temporal consistency-based re-labeling is set as six. Each label contains $K = 5$ sequential ego-relative world coordinates, and the interval between each waypoint is 0.5 second. Thresholds ϵ_a and ϵ_b are set as 0.5. When training the planning model, we train over a mix of the recent incoming data and the experience replay, as shown in Alg. 1 of the main paper. We leverage multiple data augmentation strategies including randomized brightness, optical distortion, dropout, and Gaussian blur. The model inference is highly efficient, with 50Hz and 100Hz for the inverse dynamics and planning policy models, respectively, on an NVIDIA RTX 3090 GPU (ensemble inference can be parallelized). Training of our agent for the results in the paper takes approximately three days.

Ensuring Ensemble Diversity: Our replay buffer incorporates encountered samples with the highest epistemic uncertainty, as estimated via ensemble disagreement. The success of this selection mechanism pivots on maintaining a diverse set of models, often achieved by using different initial random seeds for the models. We have experimented with additional diversity-promoting and calibration mechanisms [13, 15, 16], but did not observe a benefit over simple variation in the random seed initialization (we note that most prior work studies such ensemble-based disagreement in simpler settings with minimally complex, noisy, and OOD regression scenarios). Instead, we employ a simple strategy that ensures maintaining model diversity throughout the self-training process. Each inverse dynamics model only observes their own generated pseudo-labels, and *reservoir sampling* [18] is employed to make sure that each sample in the recent video stream has an equal probability of being selected into the dataset. We did not find it beneficial to further optimize the ensemble training, e.g., using a dedicated replay buffer, and thus rely purely on the random seed and distinct pseudo-labels to maintain a diverse ensemble. Leveraging additional models in the ensemble beyond five is likely to result in further gains (as we plan to study in the future), yet leads to added computational overhead.

1.3. Baselines

In Sec. 2, we validate our underlying the role of leveraging an inverse dynamics model for pseudo-labeling over two recent methods based on pre-training (PPGeo [20]) and policy self-training directly (SelfD [21]). We select PPGeo due to its state-of-the-art performance compared to prior works (e.g., [22]), and leverage the publicly available authors' implementation. However, as the aforementioned methods make minimal use of underlying demonstrations in the videos and are not trained in an incremental manner, they are complementary to our proposed framework. Below, we discuss the relevant baselines, yet highlight that prior work in incremental learning do not generally incorporate learning from informative but highly-uncertain samples which are pseudo-labeled nor a complex real-world driving decision-making task.

Dark Experience Replay (DER): DER [5] is a general continual learning framework that supports tasks boundaries blur and domain shift settings. However, DER leverages reservoir sampling-based buffer (which does not effectively manage informative samples) and has only been previously studied in simplified and fully supervised classification settings (on an MNIST-based benchmark).

Rainbow Memory: Rainbow memory [2] is a recent method that tackles blurry task boundaries and integrates a diverse memory management strategy based on per sample uncertainty. While relevant to our buffer construction strategy, such methods for selecting high-uncertainty samples have only been previously studied in fully supervised settings with clean ground-truth labels. In contrast, due to the inherent noise in the pseudo-labeling and self-training process, selecting high-uncertainty samples in our settings can be counterproductive, and must require careful handling. Moreover, uncertainty is measured based on data augmentation, which we find to be unreliable in our settings (we leverage model ensemble-based disagreement).

PuriDivER: PuriDivER is closely related to us since they operate on noisy data streams, while incrementally learning on several datasets, including CIFAR-10, CIFAR-100, Food-101N, ImageNet, and Webvision [3]. Similarly to PuriDivER, we leverage a two-cluster Gaussian Mixture Model (GMM) in order to perform re-labeling and loss-based filtering of samples. However, PuriDivER assumes access to both noisy and clean ground-truth labels throughout the re-labeling process. In contrast, our settings involve *pseudo-labels* with potentially significant and diverse noise characteristics. As will be further discussed in Sec. 2, as PuriDivER assumes access to a clean set of labels the employed re-labeling and consistency regularization strategies (e.g., based on AutoAugment [11]) did not generalize at all to our settings since consistency regularization training and re-labeling are less reliable due to the inherent noise in all the pseudo-labels. Instead, our framework leverages temporal consistency-based re-labeling.

1.4. Web Data Statistics

To validate our proposed incremental self-training policy, we download a large set (totaling 369 videos) of driving footage from YouTube from all over the world (30 countries based on the metadata). For practical reasons, we focus on longer dashcam videos in our main results. All frames are resized to an image size of 400×225. As mentioned in the main paper, the set incorporates diverse driving scenes, including urban, highways, rural roads, off-road trails, mountains, forests, and coastal areas as well as diverse weathers and times of day. Examples are shown in Figure 1. While the dataset is already large, every day hours of driving data are uploaded to public sources. Our never-ending learning system is specifically designed to efficiently train from additional videos in the future (the model has thus far seen over ten million frames, and counting). We plan to continue and test the limits of our proposed approach on unconstrained and highly diverse real-world video data.

1.5. Pseudo-labels Transformation

We describe the details of transformation $\mathcal{T}$ here (Eq. 4 in the main paper). Given image $\mathbf{I}_t$, to compute the future waypoints of $\mathbf{I}_t$, we first estimate the ego-pose of the image pairs $\{(\mathbf{I}_i, \mathbf{I}_{i+1})\}_{i=t}^{t+h}$. In our setting, we are computing the next five future waypoints, the interval between waypoints is 0.5 second, and the fps of extracted frames from videos are 10Hz, thus h is set as 25. We set the location of image $\mathbf{I}_t$ as the coordinate origin. The pose of each subsequent image $\mathbf{I}_n$ with respect to $\mathbf{I}_t$ can be recovered by integrating estimated relative poses as $\prod_{i=t}^{n-1} \hat{\mathbf{p}}_i$ where $\hat{\mathbf{p}}_i = \frac{1}{M} \sum_{m=1}^{M} f_{\psi_m}^{\text{inv}}(\mathbf{I}_i, \mathbf{I}_{i+1})$ are the averaged estimated ego-motion from the ensemble models. Once we have computed the pose of each subsequent image with respect to $\mathbf{I}_t$, waypoint can be derived from the translation of the pose, speed is calculated as the average velocity between the origin and the first waypoint, command is similarly recovered through thresholding the estimated trajectory sequence of five waypoints to determine the turn direction.

2. Detailed Analysis and Ablations

In this section, we provide additional baseline comparisons (Sec. 2.1), ablation studies (Sec. 2.2), supportive validation in closed-loop evaluation (Sec. 2.3), as well as additional qualitative examples (Sec. 2.4).

2.1. Baseline Comparisons

Table 1 compares our driving policy against recently proposed non-continual frameworks (SelfD [21] and PPGeo [20]). We note that as SelfD have not released their code, we re-implement their method to the best of our efforts. For PPGeo, we utilize the public GitHub repository and the pre-trained visual encoder. All models assume access to nuScenes-Boston. As SelfD and PPGeo are not incremental learning frameworks, we show the ADE (Average Displacement Error) for these models on our multi-dataset evaluation in Table 1. We further show an ablation baseline in which the inverse dynamics model is fixed following the initial training phase, i.e., the inverse dynamics teacher doesn't continuously update with the policy student model. In all cases, the proposed approach outperforms the baselines in generalization. We find the geometric pre-training method of PPGeo to be brittle on our large and diverse dataset. Moreover, while the underlying image representations may be robust the method cannot effectively leverage decision-making from the unlabeled data. We find it to perform poorly on our harsh cross-dataset generalization settings. Our model also outperforms both PPGeo and SelfD, with over 42% reduction in ADE. We also highlight the benefit of continuously updating both teacher and student models over pseudo-labeled data, i.e., as opposed to freezing inverse dynamics teacher during training. This experiment demonstrates that continually adapting both models effectively generalizes over out-of-distribution data.

2.2. Ablations on Model Components

Our model comprises a complete agent that can handle highly diverse samples and noisy pseudo-labels during training. In this section, we demonstrate the contribution of our proposed temporal consistency-based re-labeling method and adaptive

Table 1. **Validation of Policy Training Strategy in Open Web Settings**. We compare the performance of our semi-supervised learning pipeline, where we train a driving policy using pseudo-labels from an ensemble inverse dynamics models, against two baselines, PPGeo [20] and SelfD [21] (please refer to Sec. 2.1). We also compare with the ablation baseline that trains an incremental self-training policy student with fixed inverse dynamics teacher. Remarkably, through our buffer selection and re-labeling process, our proposed approach outperforms all prior methods.

Methods	$\bar{L}_{-1}$	$\bar{F}_{-1}$	ADE_{-1}
PPGeo [20]	/	/	5.748
SelfD [21]	/	/	1.962
∞**-Driver** (Teacher fixed)	1.929	-0.146	1.734
∞**-Driver (Ours)**	**1.174**	**-0.017**	**1.130**

filtering mechanisms. We show the ablation results in Table 2, where we find the individual components to be synergistic for handling noise in pseudo-labels. We leverage three continual learning evaluation metrics: the revised Average Loss $\bar{L}_{-1}$, Forgetting $\bar{F}_{-1}$, and ADE_{-1} after the model incrementally trains on ten million YouTube images. These metrics provide insights into evaluating the performance and efficacy of our model under different conditions and configurations. Without temporal consistency-based re-labeling and adaptive filtering mechanisms, we find the buffer to accumulate a high proportion of noisy samples. This increase of noisy samples in the buffer results in less precise waypoints prediction and leads to drastic ADE fluctuations, as evidenced by the higher $\bar{L}_{-1}$ and ADE_{-1}.

Adaptive Filter Mechanisms: Our datasets contain ample amounts of "hard" examples, i.e., examples that are difficult for the model to learn from or re-label. To effectively learn under such challenging settings, we further implement two additional filtering mechanisms that can remove samples that hinder the self-training process. Specifically, we implement a confidence-based filter in the model, i.e., for the removal of samples with deprecated images by identifying and rejecting those with abnormally high entropy. We also implement a low-loss-based filter based on Eq. 9 of the main paper (based on the planner loss in Eq. 5), i.e., samples that exhibit abnormally large ADE scores are deemed to be with highly noisy pseudo-labels and are removed prior to updating the buffer. We observe the improvement in both the revised Average Loss and ADE score in Table 2 which indicates the benefits brought by a buffer with more accurate pseudo-labels. As training progresses, our buffer can be progressively replenished with new incoming driving data with better pseudo-label quality.

Temporal Consistency Module: To minimize removal of samples and effectively make use of diverse data, we incorporate a temporal consistency-based re-labeling mechanism. Our temporal consistency-based re-labeling module enables pseudo-label purification by utilizing temporal information to refine and improve the quality of pseudo-labels. We note that prior methods in continual learning may leverage single-frame strategies for sample re-labeling and purification, e.g., based on a convex combination of pseudo-labels or consistency regularization from data augmentation [3, 4]. Yet, we found these to not be beneficial due to the significant noise in the pseudo-labels. The improvement due to the introduced temporal consistency (TC) module is particularly evident in the Forgetting measure and the ADE score. These improvements demonstrate the effectiveness of incorporating temporal consistency in the relabeling process, leading to more accurate and reliable training samples, which in turn positively affects the model's overall performance, especially in its ability to retain learned information and predict more precise waypoints. Finally, by incorporating both temporal consistency-based re-labeling and adaptive filtering mechanisms into our model, we achieve the best performance in both the Forgetting measure and the ADE score, as shown in Table 2.

2.3. Validation in Closed-Loop Settings

Open-loop metrics measure adherence to human-preferred trajectories, yet cannot fully account for errors that lead to closed-loop failure, e.g., collisions and infractions. Thus, to further validate the generalization of our findings, we validate model results in closed-loop settings using CARLA (shown in Table 3). To replicate our training scheme in simulation, we first train the visual odometry teacher model solely over one hour of driving data from routes in Town 1 and incrementally train the driving policy student in Town 2 and Town 5 (one hour of driving data each). We input the predicted waypoints to a PID controller to obtain low-level control. We further evaluate generalization performance on routes from unseen towns, specifically Towns 3, 4, 6, and 7. In order to test the agent's adaptability to diverse unseen situations, we further randomize weather conditions and front camera orientation. We leverage the standard metrics used in the Carla leaderboard [1]: success rate (SR), route completion (RC), infraction score (IS), and driving score (DS) as our metrics. Table 3 demonstrates our

Table 2. **Ablation Study Over Three Main Model Components**. We demonstrate the benefits of the two adaptive filtering mechanisms, the confidence-based filtering (**CF**, Eq. 8 of the main paper, thresholding the estimated entropy of the distribution in Eq. 2) and the planning loss-based filtering (**LF**, Eq. 9 of the main paper). We also analyze the impact of the temporal consistency-based re-labeling mechanism (**TC**) across the revised Average Loss (denoted as $\bar{L}_{-1}$), Forgetting (denoted as $\bar{F}_{-1}$), and ADE_{-1} evaluation metrics. We find the three components to be synergistic for handling noise in pseudo-labels.

EF	LF	TC	$\bar{L}_{-1}$	$\bar{F}_{-1}$	ADE_{-1}
			1.195	-0.019	1.143
✓			1.175	-0.032	1.133
	✓		1.168	-0.040	1.138
		✓	1.179	-0.017	1.134
✓	✓	✓	1.173	**-0.017**	**1.130**

Table 3. **Incremental Learning and Generalization Validation in Closed-Loop Settings.** We report closed-loop metrics of Success Rate (SR), Route Completion (RC), Infraction Score (IS), and Driving Score (DS) across incremental learning methods on CARLA. All metrics are the higher, the better. Please see Sec. 2.3 for the adaptation and generalization evaluation setup.

Metrics	**DS**	**SR**	**RC**	**IS**
PuriDivER [3]	0.33	0.07	0.49	0.63
Rainbow [2]	0.38	0.13	0.52	0.66
DER [5]	0.41	0.10	0.58	0.66
∞-Driver (Ours)	**0.47**	**0.21**	**0.61**	**0.70**

performance compared to baseline incremental learning approaches within the challenging evaluation settings. Specifically, we observe our agent to outperform the nearest baseline, DER, by about 15% in DS.

2.4. Qualitative Examples

We provide several qualitative results, including samples from our high-uncertainty replay buffer (Fig. 1), the impact of temporal consistency (Fig. 2), examples detected through the adaptive filtering mechanisms (Fig. 3 and Fig. 4), and success and failure motion planning cases (Fig. 5 and Fig. 6).

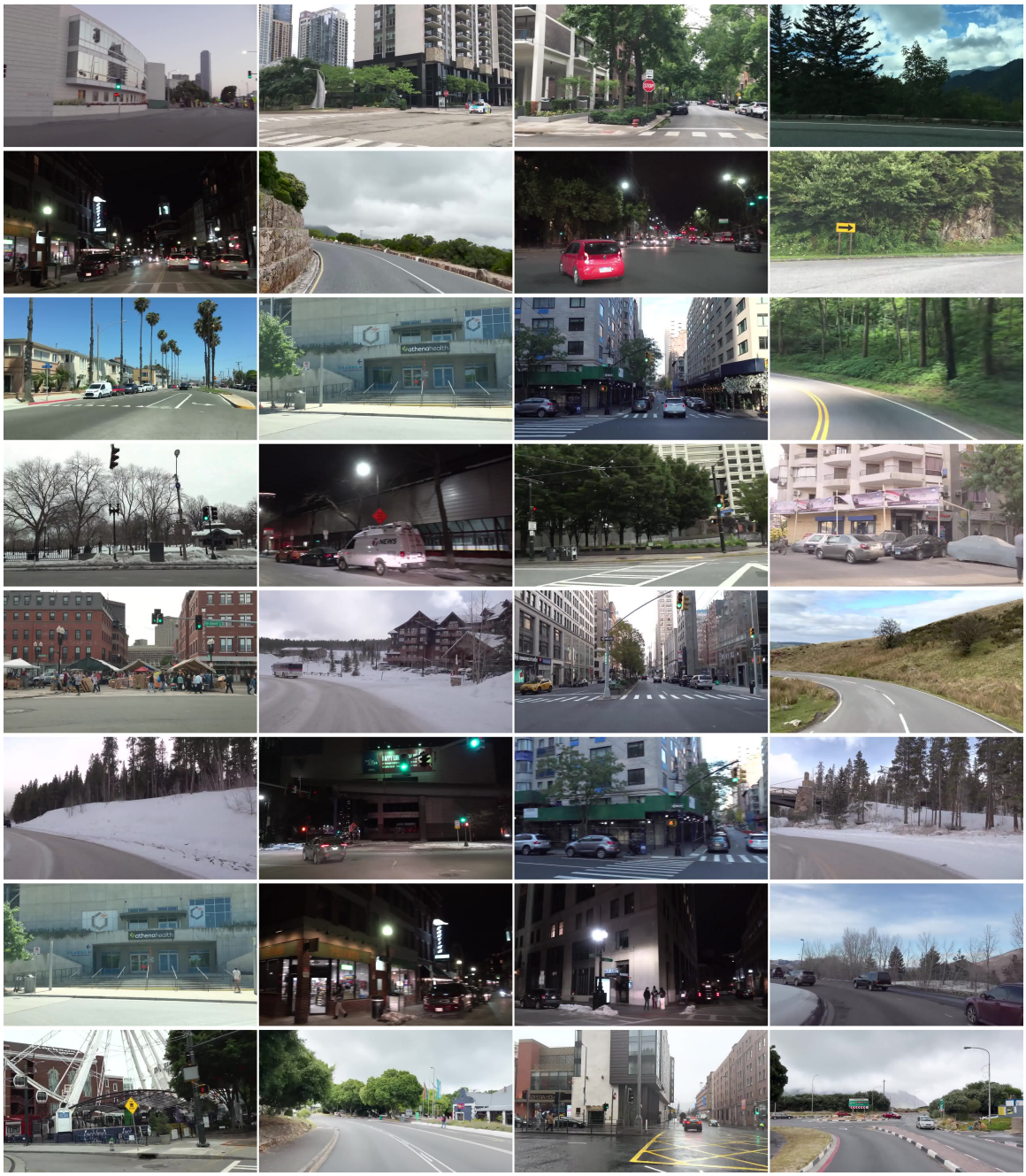

Figure 1. **Selected High-Uncertainty Buffer Samples.** We visualize example frames from the final episodic memory, demonstrating high diversity and coverage over situations, times of day, and environmental conditions.

References

[1] Carla autonomous driving leaderboard. *https://leaderboard.carla.org/*, 2022. 4

[2] Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. Rainbow memory: Continual learning with a memory of diverse samples. In *CVPR*, 2021. 2, 5

[3] Jihwan Bang, Hyunseo Koh, Seulki Park, Hwanjun Song, Jung-Woo Ha, and Jonghyun Choi. Online continual learning on a contaminated data stream with blurry task boundaries. In *CVPR*, 2022. 3, 4, 5

[4] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic

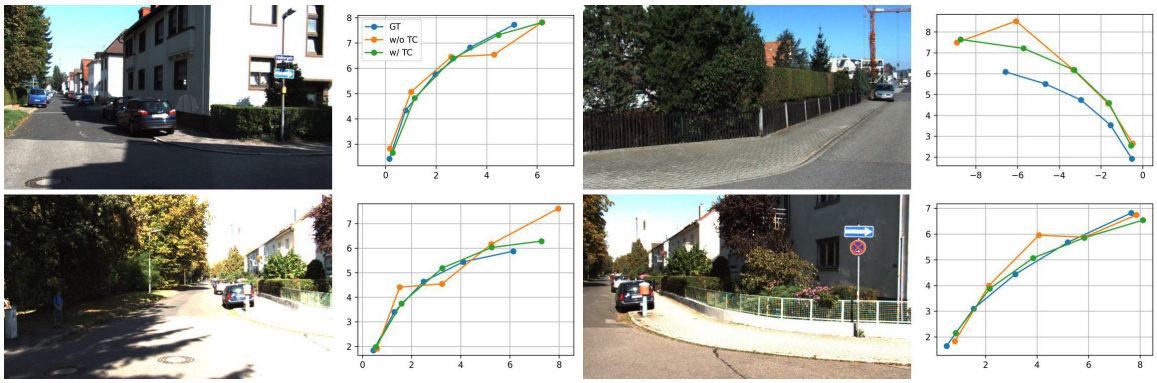

Figure 2. **Examples for the Temporal Consistency-based Sample Re-labeling.** We observe a reduction in pseudo-label noise due to temporal consistency-based re-labeling. In **blue** we show the **ground-truth** trajectory, in **orange** the original **pseudo-label**, and in **green** is the trajectory after re-labeling based on temporal consistency. We note that x-axis (lateral) and y-axis (longitudinal) **units are in meters**.

Figure 3. **Confidence-based Filter Removes Ambiguous Samples.** The confidence-based filter (Eq. 9 of the main paper) automatically detects scenarios for removal, including frame transitions in videos, conditions with image artifacts, and non-informative samples.

approach to semi-supervised learning. In *NeurIPS*, 2019. 4

[5] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *NeurIPS*, 2020. 2, 5

[6] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *T-RO*, 2021. 1

[7] Dian Chen and Philipp Krähenbühl. Learning from all vehicles. In *CVPR*, 2022. 1

[8] Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *CoRL*, 2020. 1, 2

[9] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *PAMI*, 2022.

[10] Felipe Codevilla, Eder Santana, Antonio M López, and Adrien Gaidon. Exploring the limitations of behavior cloning for autonomous driving. In *ICCV*, 2019. 1

[11] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *CVPR*, 2019. 3

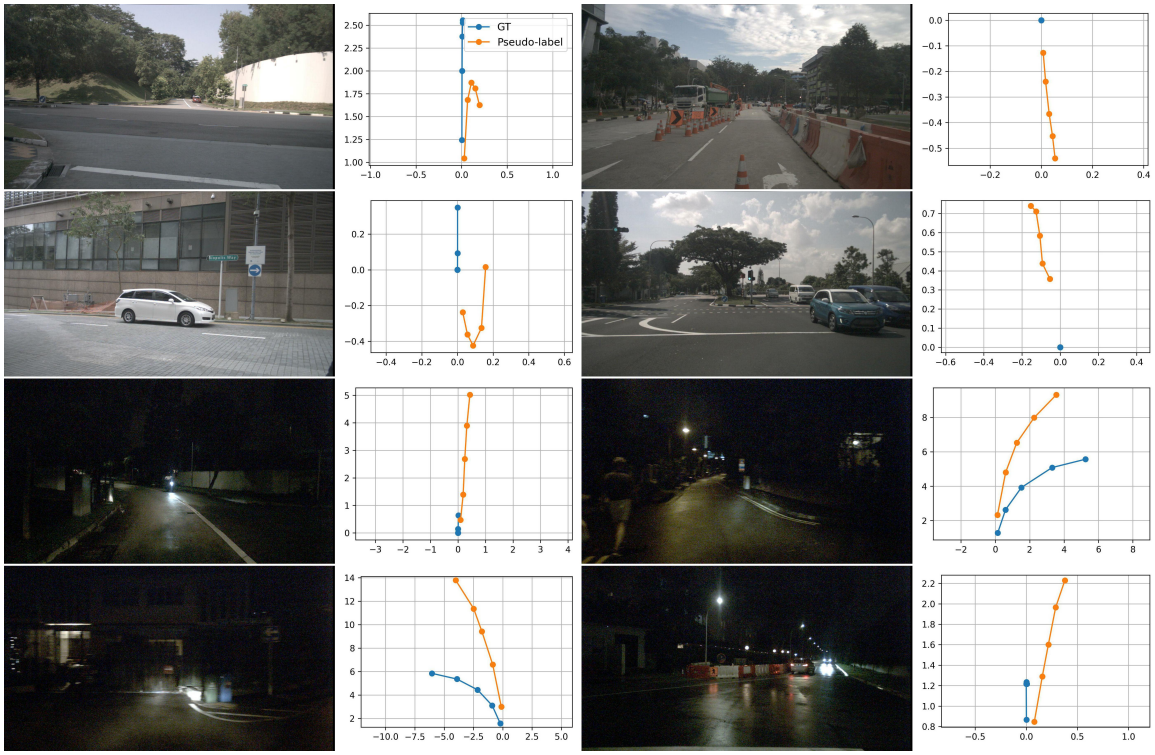

Figure 4. **Planning Loss-based Filter Removes Ambiguous Samples.** Further removal of poor and ambiguous pseudo-labeled samples using the planning loss-based filter (Eq. 10 of the main paper). In **blue** is the **ground-truth** reference trajectory, and in **orange** the computed **pseudo-label**. We note that x-axis (lateral) and y-axis (longitudinal) **units are in meters**.

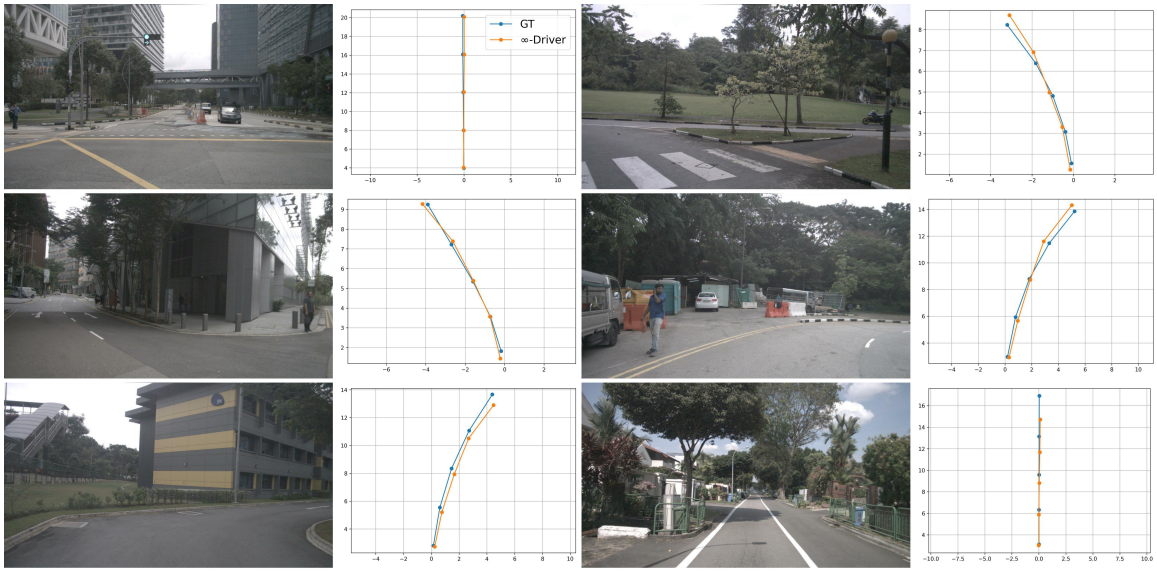

Figure 5. **Successful Planning Results.** We plot cases where the proposed **agent** (in **orange**) successfully predicts the **human-driven path** (in **blue**). We note that x-axis (lateral) and y-axis (longitudinal) **units are in meters**.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[13] Yoonho Lee, Huaxiu Yao, and Chelsea Finn. Diversify and disambiguate: Out-of-distribution robustness via disagreement. In *ICLR*, 2022. 2

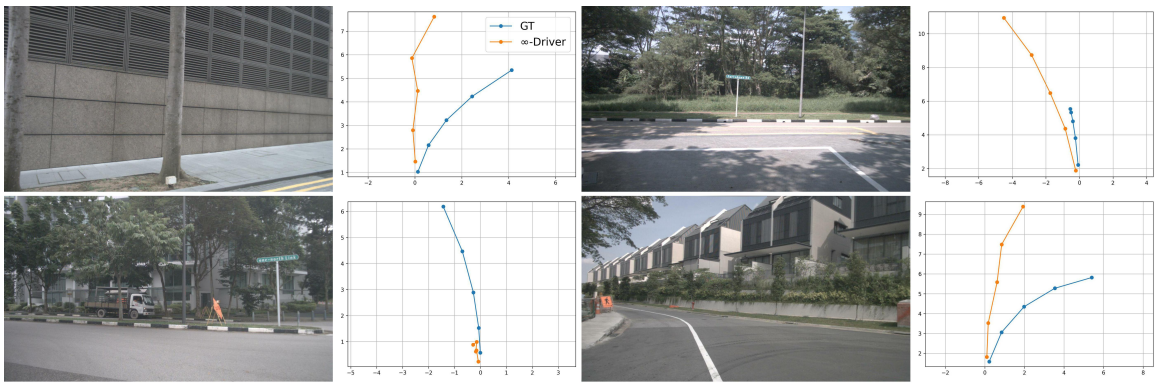

Figure 6. **Failure Cases.** We plot challenging cases where the proposed **agent** (in **orange**) fails to correctly predict the **human-driven path** (in **blue**), specifically around minimal visibility (during a turn) and non-ordinary road layouts and ego-vehicle positioning. We note that x-axis (lateral) and y-axis (longitudinal) **units are in meters**.

[14] David Mohlin, Josephine Sullivan, and Gérald Bianchi. Probabilistic orientation estimation with matrix fisher distributions. In *NeurIPS*, 2020. 1

[15] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua V Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *NeurIPS*, 2019. 2

[16] Matteo Pagliardini, Martin Jaggi, François Fleuret, and Sai Praneeth Karimireddy. Agree to disagree: Diversity through disagreement for better transferability. *ICLR*, 2023. 2

[17] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *NeurIPS*, 2021. 1

[18] Jeffrey S Vitter. Random sampling with a reservoir. *TOMS*, 1985. 2

[19] Sen Wang, Ronald Clark, Hongkai Wen, and Niki Trigoni. Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *ICRA*, 2017. 1

[20] Penghao Wu, Li Chen, Hongyang Li, Xiaosong Jia, Junchi Yan, and Yu Qiao. Policy pre-training for end-to-end autonomous driving via self-supervised geometric modeling. *ICLR*, 2023. 1, 2, 3, 4

[21] Jimuyang Zhang, Ruizhao Zhu, and Eshed Ohn-Bar. SelfD: self-learning large-scale driving policies from the web. In *CVPR*, 2022. 1, 2, 3, 4

[22] Qihang Zhang, Zhenghao Peng, and Bolei Zhou. Learning to drive by watching youtube videos: Action-conditioned contrastive policy pretraining. In *ECCV*, 2022. 2