

Supplementary Material for Learning to Drive Anywhere via Regional Channel Attention

Anonymous Author(s)

Affiliation

Address

email

Abstract: In this supplementary document, we provide additional implementation details, including network architecture and training protocol, as well as additional analysis, including ablative studies, results on CARLA, and additional qualitative examples. Qualitative results can also be seen in our supplementary video.

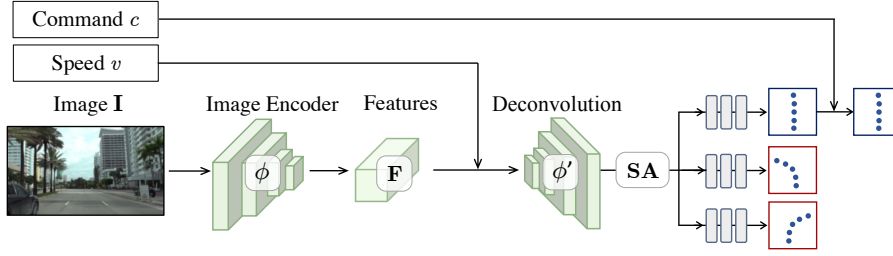
1 Implementation Details

This section first provides details regarding our proposed network architecture and discuss differences with baseline models (Sec. 1.1). Next, we provide details regarding the processing of the driving datasets to construct the multi-city benchmark used throughout the analysis (Sec. 1.2). Finally, we discuss evaluation settings (Sec. 1.3) and training protocol (Sec. 1.4).

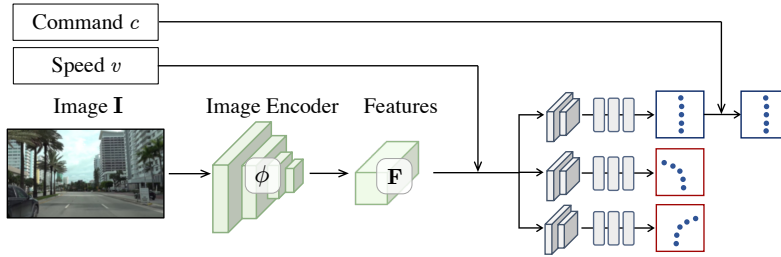
1.1 Architecture and Baselines

We leverage an ImageNet-pretrained ResNet-34 [1] as our backbone ϕ . All images are resized to 400×225 prior to being inputted to the model. To leverage diverse camera viewpoints, we build on prior models in conditional imitation learning [2, 3] and train a direct image-to-BEV prediction model, i.e., without assuming a fixed known BEV perspective transform. Moreover, we also find the removal intermediate image-level heatmaps [2, 3] (and directly regressing the BEV waypoints) to improve model performance. Fig. 1 compares our proposed network architecture for image-to-BEV planning to a standard baseline architecture (e.g., [3, 2]). Prior image-based models may utilize deconvolutional layers to obtain an image-aligned heatmap and followed by a soft-argmax ('SA' in Fig. 1) and 2D waypoint projection to the BEV space. The projection can be implemented either using a homography (i.e., known extrinsic parameters [2, 4, 5]) or with a learned projection layer [3]. In contrast, we find it beneficial to remove the intermediate image-level processing and directly predict BEV waypoints, as shown in Fig. 1(b). We replace the upsampling layers with a 3×3 convolutional layer which fuses the image and speed-based features prior to inputting to three fully-connected final prediction layers. By removing unnecessary processing steps and enabling more expressive image-to-BEV mappings, the proposed planner architecture improves from 2.45 FDE to 2.17 FDE. This improvement comes with a minimal gain in parameters (23.8M vs. 24.1M). Incorporating the geo-conditional module with three adapters further improves trajectory prediction performance to 1.93 FDE with a 24.2M parameter model. We do not find it necessary to leverage more complex, e.g., GRU-based [5, 4], prediction heads.

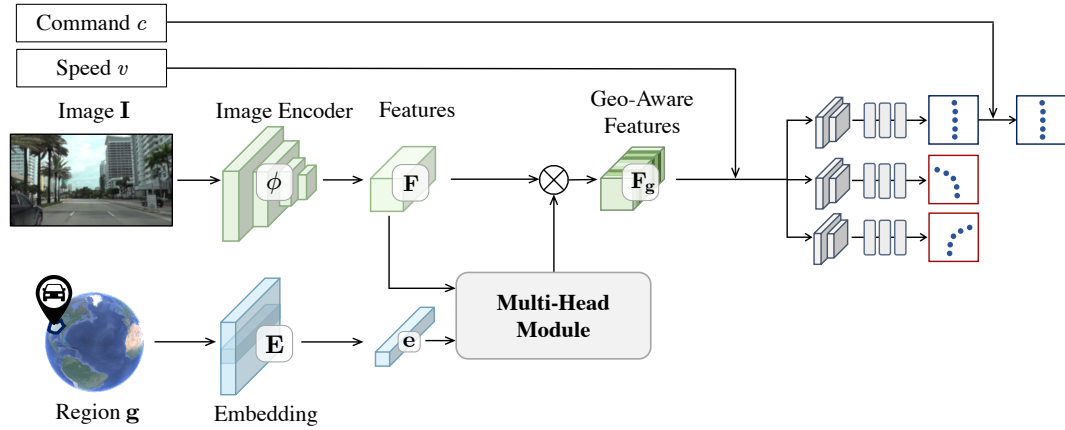
Baselines: While related methods are often studied in simulation [4, 2], we provide several baselines by following publicly available implementations. In particular, we leverage the state-of-the-art monocular agent TCP [4]. To ensure meaningful comparison with single-frame models, we also remove the temporal refinement module. As TCP leverages a control prediction branch in addition to the waypoint prediction branch, we normalize the raw control signals among the different vehicle platforms across the multiple datasets to $[0, 1]$. When comparing with the semi-supervised learning scheme of SelfD [3], we leverage a 10-hour YouTube driving dataset with available city



(a) The baseline **BEV Planner** [3] relies on alignment with the image input through upsampling (i.e., to obtain a waypoint heatmap), followed by soft-argmax (SA) [2].



(b) The **proposed planner** model simplifies the architecture such that each command branch directly predicts BEV waypoints without intermediate upsampling, 2D heatmap, or soft-argmax.



(c) The complete GeCo architecture with geo-aware feature modulation.

Figure 1: Comparing Network Architectures. As discussed in the main paper (Sec. 3.4 and Sec. 4.1), our proposed planner architecture abandons the intermediate image-aligned heatmaps and consequent soft-argmax employed by several prior works [2, 3] (top figure) and directly predicts waypoints (middle figure). The proposed architecture improves BEV waypoint prediction results by 12.9% FDE (2.45 vs. 2.17 FDE, shown in Table 1 of the main paper). Incorporating a geo-conditional module (bottom figure) further boosts performance by 4.1% (2.08 FDE) and 11.1% (1.93 FDE) without and with the proposed loss function, respectively.

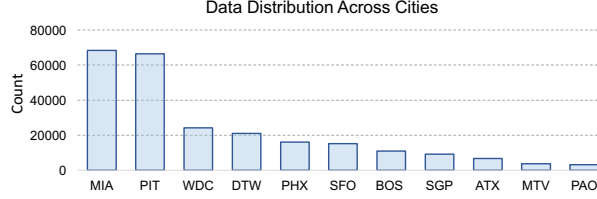


Figure 2: **Imbalanced Training Data Distribution.** Our training data is unevenly distributed across different cities, as often is the case in real-world data (y-axis is sample count).

38 taken within the 11 cities leveraged in our experiments. Subsequently, we mix the datasets to train a
 39 semi-supervised GeCo model, which is then evaluated over our multi-city benchmark.

40 1.2 Data Processing and Distributions

41 To obtain BEV waypoints for training, we standardize formats across three datasets, Argoverse 2
 42 (AV2) [6], nuScenes (nS) [7] and Waymo (Waymo) [8]. The datasets provide post-processed world
 43 coordinates for each frame obtained from GPS and other mounted sensors, e.g., LiDAR [7]. We
 44 leverage these reported ego-poses to generate a waypoint prediction benchmark. For each frame,
 45 we use the global coordinate as an intermediate to get relative positions of the future 2.5s as ground
 46 truth waypoints. The conditional command (left, forward, or right) is inferred in a semi-automatic
 47 process. First, we extract the preliminary command by thresholding the curvature of the trajectory.
 48 However, this process cannot detect subtle maneuvers, such as lane changes, which are included in
 49 our dataset. Consequently, we manually verify and annotate the initial automatic command predic-
 50 tions. For our geo-conditional module, we do not require accurate GPS information as we quantize
 51 each latitude and longitude into a city-level cluster. Each dataset provides a front-view RGB image
 52 and speed (either from the raw CAN bus or from the positioning information), which are inputted
 53 into our model as observations.

54 The data spans 11 cities: Pittsburgh (PIT), Washington, DC (WDC), Miami (MIA), Austin (ATX),
 55 Palo Alto (PAO) and Detroit (DTW), Boston (BOS), Singapore (SGP), Phoenix (PHX), San Fran-
 56 cisco (SFO) and Mountain View (MTV). The data distribution across different cities is shown in
 57 Fig. 2.

58 1.3 Evaluation Metrics and Settings

59 **Open-Loop Evaluation:** Average Displacement Error (ADE) and Final Displacement Error (FDE)
 60 are standard metrics for trajectory prediction [9]. We first compute the error within each city, and
 61 then average to obtain a balanced average metric. We also generate more fine-grained analysis by
 62 providing a breakdown over 11 semantic events in the dataset, as will be further discussed in Sec. 2.
 63 The extracted events include left turns, forward command, right turns, highway driving, heavy down-
 64 town traffic, red traffic lights, stop signs, uncontrolled intersections, pedestrian crossings, rain, and
 65 construction zones.

66 **CARLA Evaluation:** The open-loop evaluation measures the distance of waypoint predictions with
 67 real-world human drivers under complex maneuvers, including yielding, merging, and irregular in-
 68 tersections. To further validate our proposed approach, we sought to evaluate GeCo in closed-loop
 69 settings where continuous predictions are made in order to navigate a vehicle to a destination along a
 70 route. While closed-loop real-world evaluation is challenging due to safety requirements, we lever-
 71 age the CARLA [10] simulator. Yet, standard CARLA evaluation does not generally involves social
 72 and regional behavior that is dynamic across towns. Subsequently, models do not currently incorpo-
 73 rate regional modeling, and GPS information is solely used to determine a command at intersections
 74 along a route. Hence, motivated by our 11-city real-world benchmark, we introduce a new bench-
 75 mark where different towns have different traffic behavior. Our benchmark is defined over Town 1,
 76 Town 2, and Town 10. We have modified Town 2 for left-hand driving and added pedestrians and

vehicles with more aggressive behaviors, e.g., with jaywalking, higher speeds, and closer proximity, to Town 10. We tune the autopilot’s controller in order to generate optimal behavior under the novel settings. When performing closed-loop evaluation in CARLA, we also compute a Driving Score (DS) [5, 11], which is a product between the route completion and a penalty based on infractions.

1.4 Training Protocol

We study the role of our proposed network for scalable deployment use-cases using three training paradigms. To ensure standardized training across both centralized and federated training, we train the model using Stochastic Gradient Descent (SGD) [12]. In **centralized training** (where all the raw observation data is shared in a single server), we use a batch size of 48 and train for 7,500 iterations. We set the initial learning rate to 1e-1, learning rate decay as 0.997, and weight decay as 1e-3. The loss hyper-parameters are set as $\lambda_c = 1e-3$, $\lambda_g = 1e-4$, $\lambda_d = 1e-4$. For **semi-supervised training** settings, model training is done in three stages. We first download a set of online videos based on their tag which provides city-level information. We then train a supervised model using the same hyper-parameters as in centralized training above, and pseudo-label the unlabeled videos. We train three models from different initial seeds to compute a confidence score (i.e., variance) for each pseudo-label and filter low-confidence predictions. Subsequently, we train our model using the original training dataset combined with the large pseudo-labeled dataset. Here, we set the initial learning rate to 1e-3 and train the model for 500 iterations. Finally, for a fair comparison between the centralized training and the **federated training** settings, we train our federated model for 1,500 synchronous communication rounds. We treat each city as its own ‘node’ or ‘device,’ but do not share the private geo-embedding with the server (i.e., we aggregate all model parameters on the server using FedAvg [13] and FedDyn [12] excluding E). For each communication round, the model is updated for five local iterations with SGD (in this manner, total iterations remain at 7,500). We further note that we remove the geo-contrastive loss term \mathcal{L}_{g-ct} in the federated learning settings (as this information is not shared among the locations). We keep all other hyper-parameters fixed throughout the training settings.

2 Additional Ablation and Results

To supplement our findings in the main paper, we discuss four additional results. First, we provide supplementary analysis in terms of FDE (corresponding to Table 2 of the main paper with ADE), context and event-based performance evaluation (Sec. 2.1). Second, we perform additional ablation studies regarding the role of the number of heads in the multi-head module, impact of GPS noise over waypoints ground-truth in training, and clusters of the neighborhood-level models (Sec. 2.2). Third, we analyze an adaptation experiment to a novel city in Sec. 2.3 and show additional qualitative waypoint prediction results in Sec. 2.5. Finally, we perform closed-loop evaluation results using the introduced CARLA benchmark.

2.1 Additional Analysis

Final Displacement Error: For completeness, we report FDE results across training paradigms and models in Table 1. While FDE is more challenging as it emphasizes long-term prediction, we observe similar trends among the models and cities compared to the complementary ADE-based analysis in the main paper. Specifically, we demonstrate GeCo to improve over our baseline even using this harsher metric, i.e., from 2.55 to 1.93 average FDE. Semi-Supervised Learning (SSL) provides further gains compared to the centralized GeCo for most cities (excluding ATX and DTW, which show slight under-performance). When compared with Federated Learning (FL), the improvement is less pronounced compared to the gains observed with ADE and FL. While some cities are shown to significantly benefit FL over CL (e.g., BOS, SGP, PHX, SFO, MTV) others do not (e.g., MIA and ATX). ATX is a small dataset with limited diversity and high speed variability. While evaluation becomes less reliable, a low-shot learning setting can also be studied to understand such challenges in the future.

Table 1: **Analyzing GeCo with Different Training Paradigms (FDE Version).** We analyze the Final Displacement Error (FDE) counterpart of Table 2 in the main paper (which shows Average Displacement Error, ADE). FDE only considers the final waypoint, while ADE considers all waypoints along the predicted route, thus providing complementary analysis. Although both metrics demonstrate similar performance trends, final waypoint prediction is a more challenging task (hence, errors are higher). We analyze the three GeCo training paradigms (CL-Centralized Learning, SSL-Semi-Supervised Learning, and FL-Federated Learning). The results show FDE across the 11 cities in our dataset. Our planner refers to the direct image-to-BEV prediction (without the geolocation information or introduced auxiliary loss terms, see middle architecture in Fig. 1).

Settings	Methods	Avg	PIT	WDC	MIA	ATX	PAO	DTW	BOS	SGP	PHX	SFO	MTV
CL	CIL [2]	2.55	2.20	2.71	2.85	2.47	3.05	2.02	1.69	2.14	3.03	3.07	2.86
	CIRL [14]	2.48	2.39	2.55	2.82	2.37	3.13	2.21	1.61	2.03	2.59	2.88	2.72
	BEV Planner [3]	2.45	2.30	2.08	2.66	2.64	3.25	2.04	1.75	2.09	2.51	2.87	2.73
	TCP [4]	2.37	2.17	2.27	2.77	2.28	3.02	1.95	1.69	1.99	2.52	2.71	2.71
	Our Planner	2.17	2.36	2.16	2.15	2.69	2.85	1.98	1.72	2.07	1.97	1.73	2.69
	GeCo	1.93	2.19	1.97	1.94	2.48	2.83	1.80	1.55	1.96	1.88	1.62	2.80
SSL	SelfD [3]	1.98	2.24	2.08	2.03	2.49	2.70	1.89	1.54	1.84	1.55	1.52	2.53
	GeCo	1.89	2.12	1.93	1.89	2.57	2.76	1.83	1.45	1.83	1.65	1.50	2.45
FL	FedAvg [13]	2.63	2.65	2.85	2.69	3.68	3.43	2.55	1.76	2.31	2.87	2.67	3.13
	FedDyn [12]	2.14	2.48	2.34	2.44	3.42	3.10	2.35	1.53	1.86	1.86	1.85	2.55
	GeCo (FedAvg)	2.23	2.51	2.21	2.12	3.52	3.40	2.04	1.58	2.09	2.43	1.97	2.79
	GeCo (FedDyn)	1.92	2.25	1.91	1.94	3.26	3.29	1.90	1.37	1.68	1.31	1.49	2.28

Table 2: **Event-Driven Analysis.** We perform separate evaluations over subsets of our total evaluation benchmark based on semantic events defined over commands and driving conditions. The settings are, in order: left turns, forward command, right turns, highway driving, heavy downtown traffic, red traffic lights, stop signs, uncontrolled intersections, pedestrian crossings, rain, and construction zones. GeCo is shown to benefit from improved robustness across conditions compared to the baseline, i.e., due to the improved modeling capacity.

Method	Metric	Left	Fwd.	Right	Hwy.	DTown	Red	Stop	Unctrl.	Cross	Rain	Constr.
Our Planner	ADE	1.60	1.01	1.43	7.66	1.12	0.66	0.63	0.77	1.36	1.37	0.76
Full GeCo		1.48	0.92	1.19	1.37	1.04	0.60	0.66	0.72	0.80	0.80	0.56
Our Planner	FDE	2.93	1.95	2.68	12.80	2.36	1.41	1.62	1.28	2.20	2.24	0.83
Full GeCo		2.75	1.85	2.30	2.40	2.20	1.05	1.62	1.22	1.44	1.44	1.30

Event-Driven Analysis: Table 2 shows a breakdown of driving performance over different events and maneuvers. Here, we see a significant benefit for the introduced regional awareness, higher modeling capacity, and more balanced contrastive objective. For instance, GeCo improves performance over the unevenly distributed commands (a ratio of 2 : 20 : 3 among left:forward:right in our dataset), for right command from 1.43 ADE with the planner and up to 1.19 ADE with GeCo. Other conditions, such as highway, downtown, crossings, and rain conditions all show improvements as well. These results suggest that the situational adapters are able to accommodate various conditions both within and across geo-locations.

2.2 Ablation Studies

Number of Heads : We investigated the impact of increasing the number of heads in Table 3 on the performance of the model in cities with different amounts of data. We observe that when cities have a sufficient amount of data (defined as more than 10,000 frames), the error rate decreases and remains consistently low as the number of heads increases. This can be attributed to the increased model capacity, allowing for more effective feature extraction and improved learning of city-specific patterns. However, when the training data is limited, increasing the number of heads can still result in worse performance on some of the cities, potentially due to overfitting the small data sample. Thus, efficiently increasing model capacity in small-data domains remains a challenge.

Table 3: **Number of Heads in the Multi-Head Module.** We vary the number of heads in the transformer model and compute the resulting model’s ADE. We observe a drop in performance beyond three heads (selected throughout the experiments). We demonstrate this to be due to the cities with small (less than 10,000) data samples which may not benefit from the increased modeling capacity.

# of Heads (H)	Large-Data Cities	Small-Data Cities	All Cities
1	1.09	1.27	1.20
2	1.00	1.28	1.15
3	0.98	1.22	1.09
5	1.01	1.28	1.13
7	1.00	1.27	1.12

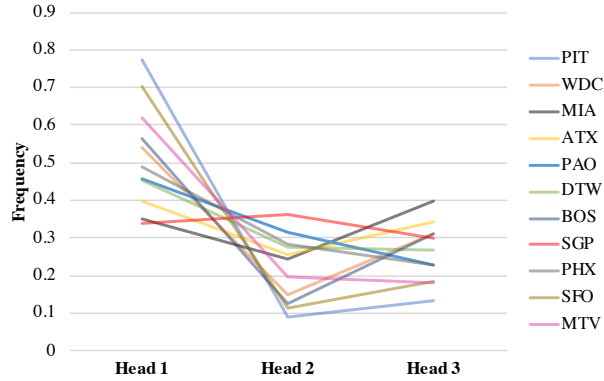


Figure 3: **Head Weight Distribution over Cities.** While the heads and their weights are learned in a data-driven manner, we do find specialization of certain heads across regions. For instance, the unique tropical scenery of Miami (MIA) gives rise to a distinctive pattern among head 1 and head 3, as well as Singapore (SGP).

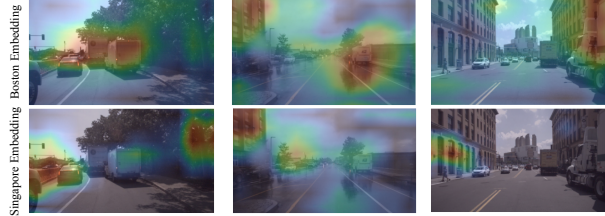


Figure 4: **Attention Visualization in the Geo-Conditional Transformer.** We visualize the attention pattern of the selected head (head with the highest weight) under the same input observations but given different region embeddings. Given Boston embeddings, the weight is shown to concentrate more on the objects in the middle and right portion of the image. While the Singapore embeddings guide attention towards the left portion of the image, which is attributable to the left-hand driving scenario in Singapore.

142 Fig. 3 studies the frequency of the head of the highest weight. Notably, our results indicate that
 143 the distribution in Singapore, where left-hand driving is practiced, differs from that of other cities.
 144 Additionally, the unique tropical scenery of Miami gives rise to a distinctive pattern as well.

145 **Effects of Geo-Conditional Transformer:** Fig. 4 shows the visual effect of the attention pattern of
 146 the selected head (head with the highest weight) under the same observations, given different region
 147 embeddings. Given Boston embeddings, the head concentrates more on the objects in the middle
 148 and right portion of the image. While the Singapore embeddings guide attention towards the left
 149 portion of the image, which is attributable to the left-hand driving scenario in Singapore. We note

Table 4: **Impact of GPS Noise.** Positioning accuracy (e.g., for obtaining waypoints for training over new locations) is known to be variable. We analyze training with different degrees of Gaussian noise imposed over the ground-truth waypoints (σ standard deviation, in meters). We introduce the noise to simulate the data collection in real-world conditions, i.e., in scenarios where the model may be trained over raw GPS measurements without extensive LiDAR-based filtering performed in current autonomous driving datasets. Metrics are averaged over cities.

Noise	ADE	FDE
Original	1.05	1.93
$\sigma = 1$	1.16	2.21
$\sigma = 3$	1.28	2.39

Table 5: **Number of K-Means Clusters vs. Model Performance.** We vary the number of clusters for K-means when obtaining additional finer-grained regions (i.e., neighborhood-level clusters) for each city from publicly available GPS traces [15]. We note that this data may not always be available within all city or country regions. While the additional fine-grained information can further improve our model, we find performance to the plateau beyond three clusters for our data.

# of Clusters	PIT	WDC	MIA	ATX	PAO	DTW
1	1.15	1.27	1.60	1.13	1.65	0.96
3	1.16	1.09	1.11	1.10	1.42	0.95
10	1.21	1.06	1.09	1.22	1.39	0.98

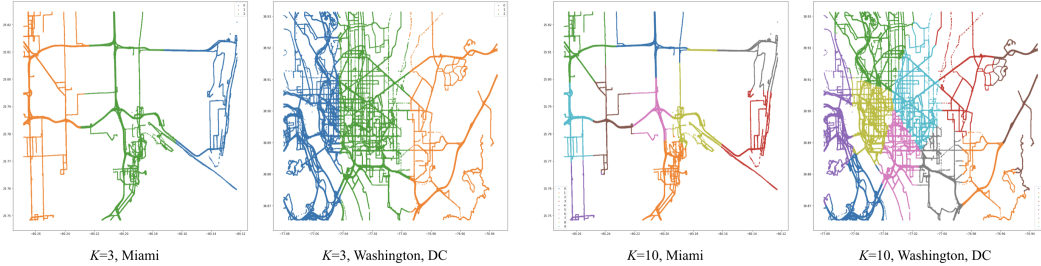


Figure 5: **Example Neighborhood Clustering by GPS Traces from OpenStreetMap.** We utilize GPS trace data from OpenStreetMap (OSM) [15] to automatically divide cities into sub-regions based on traffic patterns. We leverage K-means clustering to analyze the ability of our model to handle finer-grained regions within cities. Clustering results of Miami and Washington, D.C. with respect to the number of clusters are shown (for 3 and 10 clusters). Despite the coarse clustering, meaningful clusters emerge, e.g., Miami’s beach (blue) vs. downtown area (green) in the leftmost $K = 3$ figure.

150 that our supplementary contains additional ablations regarding number of heads in the multi-head
151 attention module and the impact of dataset size on training such higher capacity models.

152 **Ground-Truth Noise Analysis:** To understand the role of scalable real-world deployment and data
153 collection, we analyze the impact of potential GPS error in Table 4. While our analyzed datasets
154 carefully post-process the reported world coordinates, we envision GeCo deployed across more
155 diverse and potentially noisy settings. We therefore report the performance degradation due to the
156 addition of 1m and 3m Gaussian noise over the training waypoints. Specifically, we find that even
157 with added noise, GeCo obtains decent prediction performance, outperforming prior state-of-the-art
158 models that are trained with clean waypoints, e.g., 2.39 FDE with 3m noise vs. 2.45 FDE for the
159 baseline [2, 14]).

160 **Unsupervised Clustering of Regions:** In Table 5, we explore the benefit of finer-grained regional
161 clustering choices, i.e., within each city, when defining g . This experiment can uncover potential

Table 6: **Adaptation to a New City.** We fine-tune on additional data from Guangzhou, China [16]. GeCo not only outperforms the baseline model on the new city but also maintains the performance on previously seen cities.

Cities	Seen Cities (Before → After)	GZ
BEV Planner [3]	1.24 → 1.33	0.87
GeCo	1.05 → 1.04	0.84

Table 7: **Closed-Loop Evaluation in CARLA.** We report closed-loop metrics of Success Rate (SR), Route Completion (RC), Infraction Score (IS) and Driving Score (DS) compared to a baseline planner which is not trained in a geo-aware manner.

Metrics	ADE ↓	FDE ↓	SR ↑	RC ↑	IS ↑	DS ↑
BEV Planner [3]	0.58	0.97	0.29	0.50	0.61	0.36
GeCo	0.46	0.79	0.36	0.69	0.67	0.50

benefits from modeling intra-city settings. To achieve this, we utilize GPS trace data from OpenStreetMap (OSM)[15] and cluster cities into sub-regions based on traffic patterns. For example, MIA clustering results in semantic regions, e.g., downtown vs. beach areas, with large improvements in prediction performance for the finer-grained model (from 1.60 to 1.11 ADE). Similarly, WDC is clustered into downtown and highway regions, also benefiting performance (from 1.27 to 1.09 ADE). We note that while these examples suggest our model can further benefit model improvement, such clustering data may not be available for many locations, and thus cannot always be assumed.

Table 5 further explores the benefits of finer-grained clustering on GeCo model performance, i.e., within city neighborhoods. To achieve this, we employ GPS trace data from OpenStreetMap (OSM) [15] and divide cities into sub-regions based on traffic patterns via K-means clustering. Fig. 5 shows example clustering results in MIA and WDC with respect to $K = 3$ and $K = 10$ clusters. For instance, the downtown areas of both two cities can be seen as clusters for both $K = 3$ and $K = 10$. While somewhat coarse in its clustering, we find that this privileged region information (which may not always be available) can introduce additional benefits when used as \mathbf{g} during model training and evaluation.

2.3 Adaptation to a New City

In practice, a geo-aware model may be required to learn to drive in a previously unseen region with a significant domain gap. To understand the model performance of GeCo under such learning settings, we extract an additional city (Guangzhou, China) from a different dataset, ApolloScape [16]. In this case, a large domain gap occurs due to the differing social norms and traffic density in China. We mix the new city with the prior 11, and continue fine-tuning the model. Our results in Table 6 indicate that GeCo can learn to drive in the new city while also maintaining similar performance levels for the previously observed cities (i.e., without forgetting). In contrast, the baseline model, which does not incorporate the explicit geo-aware module, has higher error while also impacting performance on prior seen cities.

2.4 Experiments on CARLA

The results of the closed-loop experiments on CARLA are shown in Table 7. We find consistent improvements in success rate, route completion, and infractions (the supplementary video shows qualitative examples where the baseline model struggles with left-hand driving). We compute both open and closed-loop metrics by saving the expert actions for the test sequences. GeCo outperforms the baseline planner [3] on both open-loop metrics (reducing ADE from 0.58 to 0.46) and closed-loop metrics (improving driving score by 38%, from 0.36 to 0.50). Nonetheless, overall success rates are quite low for our benchmark, as it contains significant behavior variability. This highlights the

challenging nature of the region-aware decision-making task for current imitation learning models, either in the real-world or in simulation.

2.5 Qualitative Results

We show additional qualitative results in Fig. 6 and Fig. 7 (on the next page). As depicted in the figures, the GeCo generally provides better performance on challenging tasks of navigating across different regions and events, including turns and merging (requires reasoning over traffic directionality) as well as speed limits. Fig. 8 depicts failure cases, which show challenging conditions involving rare rule violations by other vehicles.

References

- [1] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [2] D. Chen, B. Zhou, V. Koltun, and P. Krähenbühl. Learning by cheating. In *CoRL*, 2020.
- [3] J. Zhang, R. Zhu, and E. Ohn-Bar. SelfD: Self-learning large-scale driving policies from the web. In *CVPR*, 2022.
- [4] P. Wu, X. Jia, L. Chen, J. Yan, H. Li, and Y. Qiao. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. In *NeurIPS*, 2022.
- [5] K. Chitta, A. Prakash, B. Jaeger, Z. Yu, K. Renz, and A. Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *PAMI*, 2022.
- [6] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes, et al. Argoverse 2.0: Next generation datasets for self-driving perception and forecasting. In *NeurIPS*, 2021.
- [7] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020.
- [8] P. Sun, H. Kretschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020.
- [9] K. Mangalam, H. Girase, S. Agarwal, K.-H. Lee, E. Adeli, J. Malik, and A. Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *ECCV*, 2020.
- [10] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. CARLA: An open urban driving simulator. In *CoRL*, 2017.
- [11] Carla autonomous driving leaderboard. <https://leaderboard.carla.org/>, 2022.
- [12] D. A. E. Acar, Y. Zhao, R. M. Navarro, M. Mattina, P. N. Whatmough, and V. Saligrama. Federated learning based on dynamic regularization. In *ICLR*, 2021.
- [13] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, 2017.
- [14] X. Liang, T. Wang, L. Yang, and E. Xing. CIRL: Controllable imitative reinforcement learning for vision-based self-driving. In *ECCV*, 2018.
- [15] J. Bennett. *OpenStreetMap*. Packt Publishing Ltd, 2010.
- [16] X. Huang, P. Wang, X. Cheng, D. Zhou, Q. Geng, and R. Yang. The apolloscape open dataset for autonomous driving and its application. In *TPAMI*, 2019.

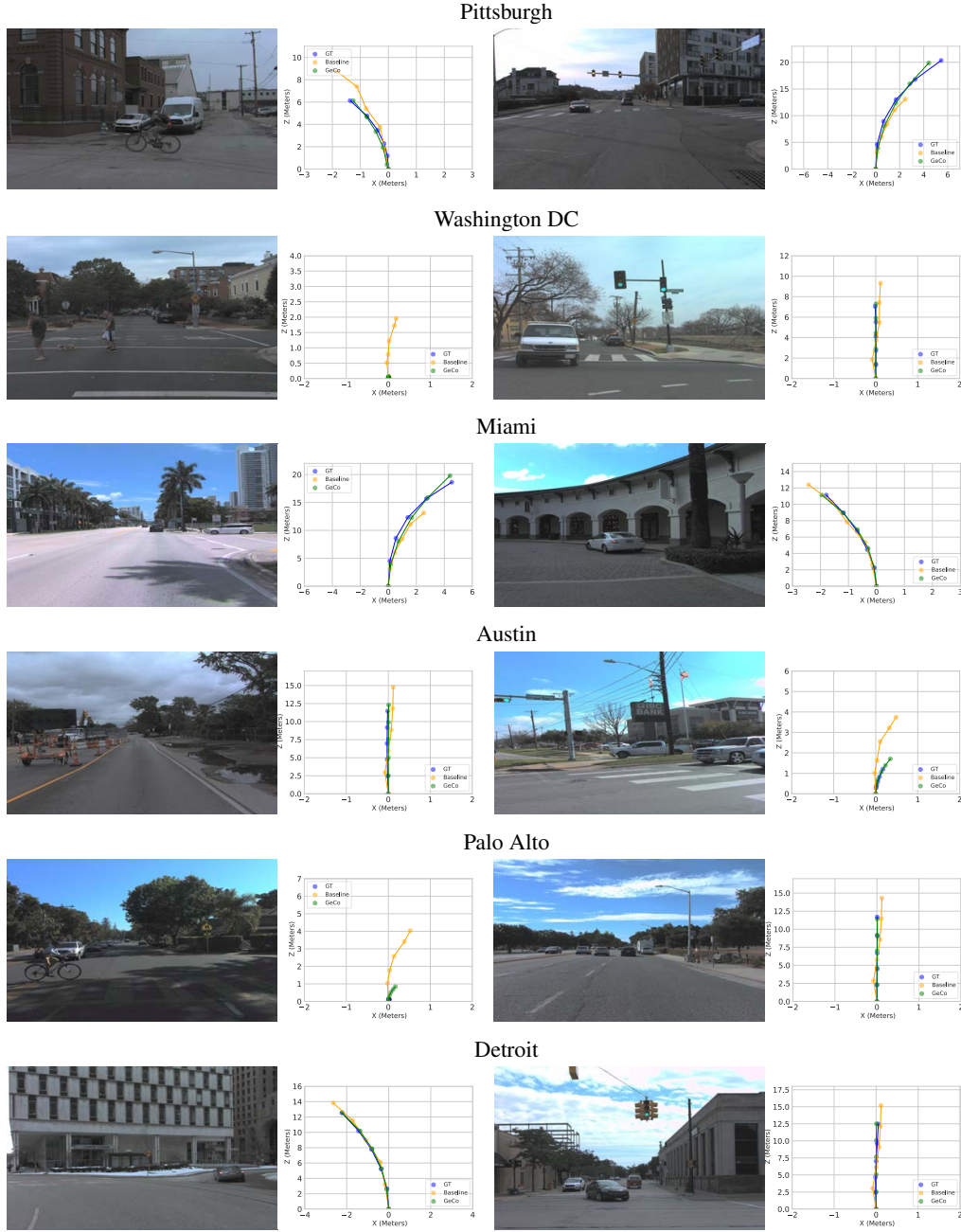


Figure 6: **Qualitative Results across Cities in Argoverse 2 Dataset.** We plot predicted waypoints in the BEV for GeCo, the ground truth trajectory, and the baseline planner model. GeCo is shown to improve reasoning over regional speed and scenarios as well as general navigation and intricate maneuvers.

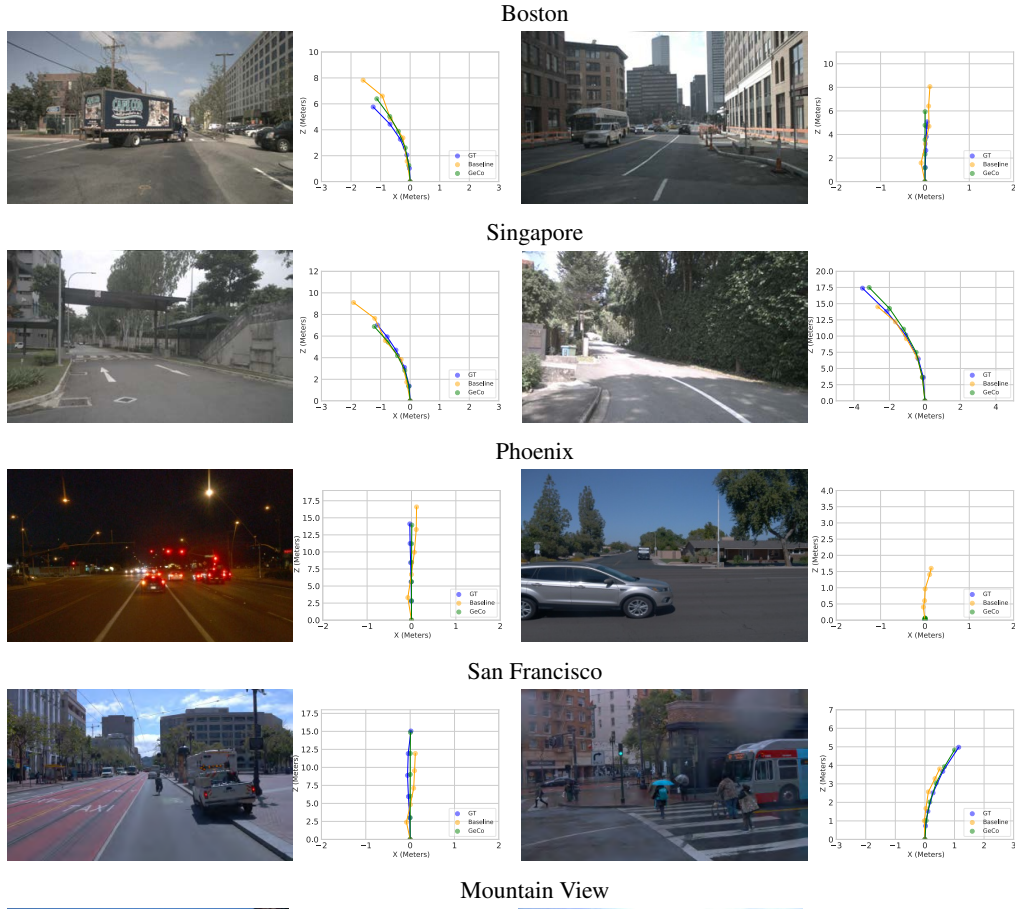


Figure 7: **Qualitative Results across Cities on nS and Waymo Dataset.** We plot predicted waypoints in the BEV for GeCo, the ground truth trajectory, and the baseline planner model. GeCo is shown to improve reasoning over regional speed and scenarios as well as general navigation and intricate maneuvers.

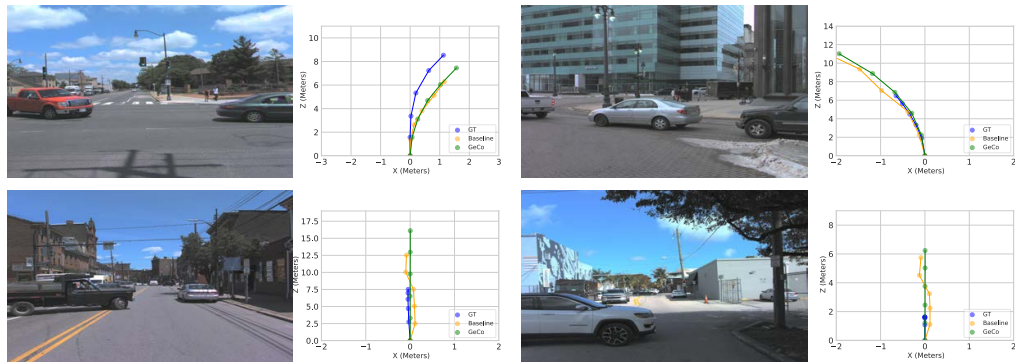


Figure 8: **Example Failure Cases.** Challenging cases where GeCo fails to produce safe driving behavior, often due to dense settings or rare behaviors.