(Supplement) - Multi-Modal Aerial-Ground Cross-View Place Recognition with Neural ODEs

S1. Supplement Experiments

S1.1. Performance with Different Aerial Models

We also test to use different and stronger backbones in the aerial models for cross-view PR. As shown in Tab. S1, performance improvements can be seen in most metrics. This shows the potential of scaling up the pipeline for more effective cross-view multi-modal PR.

In addition, even with AnyLoc [S3] as the aerial model that relies purely on DINOV2's pre-trained weights [S5] without any downstream task (PR) fine-tuning, AGPlace can still achieve considerable recall performance. This indicates that given a frozen database, the query model can still be optimized to align with the database, such that query and database descriptors can be matched. It shows the promising future of the aerial-ground multi-modal PR task in actual deployment.

Ground Model	Aerial Model Backbone	KITTI360-AG R@1	R@5	R@10
AGPlace	ResNet18	32.0	47.6	54.9
AGPlace	ResNet34	33.2	48.1	55.4
AGPlace	ResNet50	33.5	48.5	56.0
AGPlace	AnyLoc* (DINOV2-ViT-S)	25.7	38.0	44.4
AGPlace	SALAD (DINOV2-ViT-S)	**34.8**	**50.0**	**57.8**

Table S1. Aerial-ground PR results on the KITTI360-AG dataset using satellite aerial maps."*" denotes the model is frozen without fine-tuning on the PR task.

S1.2. Camera FOV

Using multiple cameras is an effective approach to expand the FOV of ground-view sensors. We evaluated our model's performance with varying numbers of cameras. As shown in Tab. S2, increasing the number of cameras consistently improves performance. Notably, even with a single camera, our model achieves nearly 80% R@5 performance, demonstrating strong potential for real-world deployment.

S1.3. Point Cloud Format

As mentioned in the main paper, unlike images, point clouds can be processed in various formats, such as voxels, BEV projection, and spherical projection. We present the performance results of these different formats in Tab. S3, where the voxel format emerges as the best performer. Notably, the spherical projection outperforms the BEV counterpart, which can be attributed to the information loss that occurs during BEV projection.

Ground Sensor	nuScenes-AG R@1	R@5
LiDAR + cam×1	73.3	84.7
LiDAR + cam×2	74.2	85.8
LiDAR + cam×6	**75.6**	**87.2**

Table S2. Performance of using a different number of cameras.

Ground Inputs	KITTI360-AG R@1	R@5	R@10
image + BEV	26.1	39.6	47.2
image + sph	28.1	44.0	51.5
image + voxel	**32.0**	**47.6**	**54.9**

Table S3. Comparison on different ground inputs. "sph" stands for the spherically projected point cloud format. "BEV" stands for the birds'-eye-view projected point cloud format.

S1.4. Salience Visualization

We provide more salience visualization as shown in Fig. S2 and Fig. S1. For ground-view inputs, the salience areas include building facades, roads, and some traffic landmarks. For aerial-view inputs, the salience areas include roads and building roofs.

For both ground and aerial inputs, the salience areas include some areas that can only be perceived in the same view and cannot be perceived in the other view. For example, some ground landmarks on the road are occluded by trees or buildings, while some building roofs cannot be seen by ground agents. This indicates that, even though the pipeline is optimized for cross-view performance, inputs from the same view (i.e. ground-ground or aerial-aerial) are also implicitly to be distinguished in the cross-view pipeline.

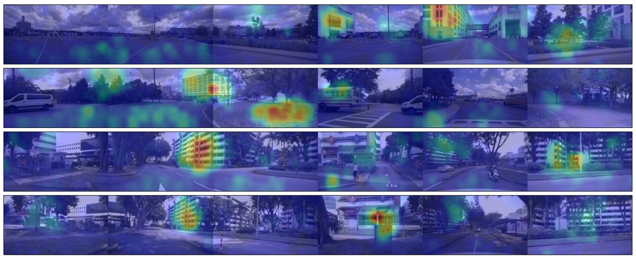

Figure S1. Ground-view salience visualization. Roads, building facades, and traffic landmarks are focused.

Figure S2. Aerial-view salience visualization. Not only roads but also building roofs are focused.

S2. More Experiment Details

S2.1. Dataset Details

The constructed KITTI360-AG and nuScenes-AG datasets are based on the vanilla KITTI360 and nuScenes respectively. Both the satellite RGB images and road maps are downloaded using Google Map Static API (settings shown in Tab. S4). The visualization of train/test ground query locations is shown Fig. S3. The dataset statistics are shown in Tab. S5.

- KITTI360. https://www.cvlibs.net/datasets/kitti-360/
- nuScenes. https://www.nuscenes.org/nuscenes
- Google Map Static API. https://developers.google.com/maps/documentation/maps-static/overview

API Key	Value
"scale"	"1"
"zoom"	"20"
"size"	"640x640"
"maptype"	"satellite" or "roadmap"

Table S4. Google Map Static API settings.

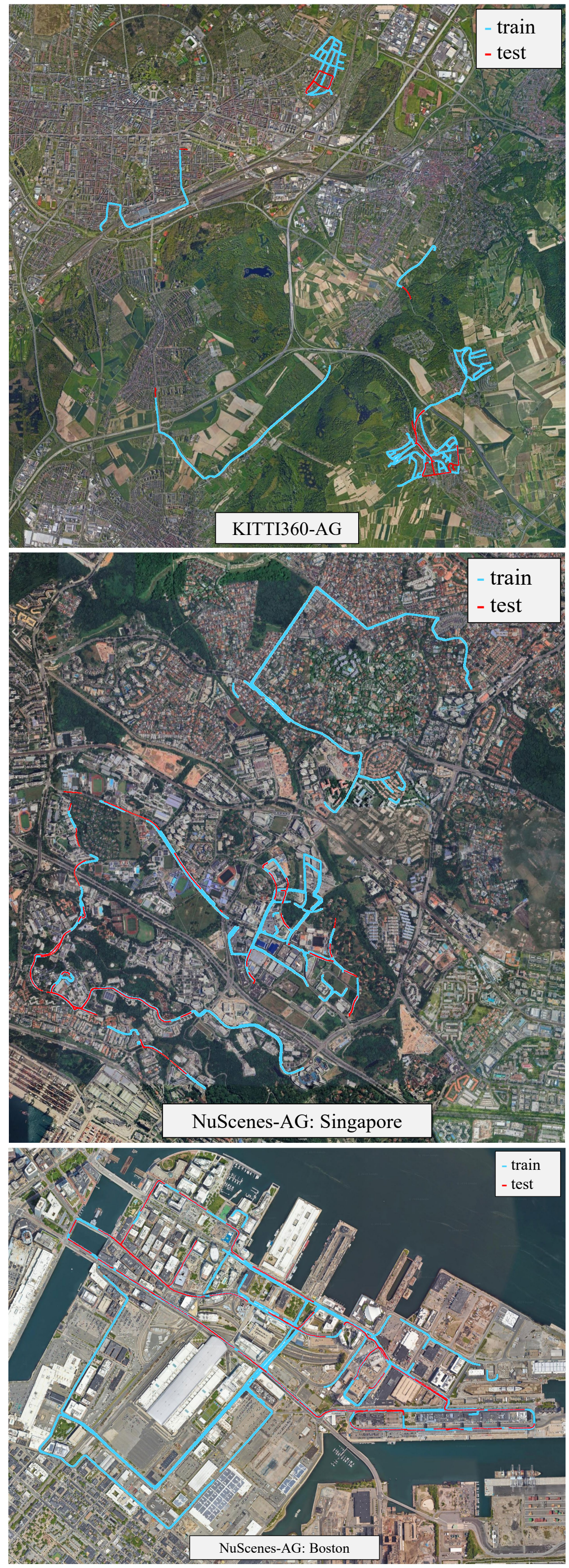

Figure S3. The query locations from train/test splits in KITTI360-AG and nuScenes-AG datasets. Both seen and unseen areas are covered.

Dataset	Train		Test	
	#Ground Q	#Aerial DB	#Ground Q	#Aerial DB
KITTI360-AG	40.6k	20.3k	7.2k	3.6k
nuScenes-AG	34.1k	38.0k	6.1k	6.7k

Table S5. Dataset statistics.

S2.2. Implementation Details

Our model is mainly built based on MinkLoc++[S4] with PyTorch and MinkowskiEngine as the frameworks. The shorter side of the input image is resized to 256. The voxel size is set as 1 m. Necessary data augmentation techniques are applied to enhance model learning. We use torchdiffeq as the neural ODE solver. The learning rates for 2D/fusion and 3D parts are set as 1e-5 and 1e-4 respectively, with Adam as the optimizer. The batch size is set as 16. The model is trained with 100 epochs. During training, the hard and soft positive thresholds are set as 10 m and 25 m respectively. During evaluation, the positive threshold is set as 25 m.

Our code is mainly based on:

- PyTorch. https://pytorch.org/
- MinkowskiEngine. https://github.com/NVIDIA/MinkowskiEngine
- MinkLoc++. https://github.com/jac99/MinkLocMultimodal
- DVGLB. https://github.com/gmberton/deep-visual-geo-localization-benchmark
- torchdiffeq. https://github.com/rtqichen/torchdiffeq

S3. Theoretical Supplement

S3.1. Proof of Theorem 1

Theorem 1. *(Non-intersection of ODE solutions.)[S2, S6] Given the ODE in (7), where f is continuous in t and globally Lipschitz continuous in γ. Let $\gamma_1(t)$ and $\gamma_2(t)$ be two solutions of the ODE in (7). If there exits initial conditions $\gamma_1(0) \neq \gamma_1(0)$, then it holds that $\gamma_1(t) \neq \gamma_2(t)$ for all $t \in [0, \infty)$.*

Proof. The detailed proof of Theorem 1 can be seen in references[S1, S7].

S3.2. Proof of Corollary 1

Corollary 1. *(Distinguished fusion states.) In each neural ODE block l, given two ODE inputs $\gamma_1^l(0) \neq \gamma_2^l(0)$ from two different scenes are different, then the two ODE outputs at this ODE block are different $\gamma_1^l(T) \neq \gamma_2^l(T)$.*

Proof. In each neural ODE block l, the neural component only consists of cascaded linear layers and activation functions, both of which are globally Lipschitz continuous in γ^l. Thus $f_{\theta^l}(\cdot)$ is globally Lipschitz continuous in γ^l. Then, since $\frac{d\gamma^l(t)}{dt} = f_{\theta^l}(\gamma^l(t))$ in (8) is an autonomous ODE system, $f_{\theta^l}(\cdot)$ is globally continuous in t. Consequently, according to Theorem 1, for each neural ODE block l, if the initial inputs are different, i.e. $\gamma_1^l(0) \neq \gamma_2^l(0)$, then the outputs at time $T \in [0, \infty)$ are different, i.e. $\gamma_1^l(T) \neq \gamma_2^l(T)$.

References

[S1] Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented neural ODEs. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[S2] Edward L Ince. *Ordinary differential equations*. Courier Corporation, 1956. 3

[S3] Nikhil Keetha, Avneesh Mishra, Jay Karhade, Krishna Murthy Jatavallabhula, Sebastian Scherer, Madhava Krishna, and Sourav Garg. Anyloc: Towards universal visual place recognition. *IEEE Robotics and Automation Letters*, 2023. 1

[S4] Jacek Komorowski, Monika Wysoczańska, and Tomasz Trzcinski. MinkLoc++: LiDAR and monocular image fusion for place recognition. In *Proceedings of the International Joint Conference on Neural Networks*, pages 1–8, 2021. 3

[S5] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1

[S6] Hanshu Yan, Jiawei Du, Vincent YF Tan, and Jiashi Feng. On robustness of neural ordinary differential equations. *arXiv preprint arXiv:1910.05513*, 2019. 3

[S7] Laurent Younes. *Shapes and diffeomorphisms*. 2010. 3