Supplementary Material for WACV 2025 'SpaGBOL: Spatial-Graph-Based Orientated Localisation'

Tavis Shore[1] Oscar Mendez[2] Simon Hadfield[1]
University of Surrey[1] Locus Robotics[2]
{t.shore, s.hadfield}@surrey.ac.uk, omendez@locusrobotics.com

1. Introduction

In this supplementary material, we expand on the information given in the paper regarding the Spatial-Graph-Based Orientated Localisation (SpaGBOL) dataset, and provide further detail on the corresponding technique's characteristics.

2. Dataset Information

The release of a novel graph-structured dataset for Cross-View Geo-localisation (CVGL) is a key contribution of SpaGBOL. We outline below how the dataset has been constructed, ensuring the work can be precisely reproduced from scratch. The process below is repeated for each city in Table 1.

Graph Data

Utilising the package *OSMnx* [1] the city graph is downloaded centred at the coordinate shown in Table 1, with width and height of 2km. The graph is simplified to only include nodes that are road junctions, all self-loops are removed from the graph, and any graphs that aren't connected to the corpus graph are discarded.

Node Data

Image data for each node is retrieved from the Google Maps API service using the packages *streetview* [2] for streetview panoramas, and *Satellite Imagery Downloader* [3] for satellite images. The latest node in each walk is denoted the target node.

Streetview Images: Each query for streetview images at a specific coordinate returns a list of historic panoramas near that point. We order potential panoramas by distance to the desired point before sampling 5 images, ensuring a wide range of capture dates. Panoramas are then horizontally cropped, removing the upper and lower 25% - these regions are overly warped, visually homogeneous, and largely unseen in their corresponding satellite image. Panoramas are all horizontally rotated to be north-aligned, simplifying yaw-aligning operations later. These are then resized to $[512, 2048]$ and stored as RGB PNG files.

Region	Central Coordinate
Brussels	(50.853481, 4.355358)
Chicago	(41.883181, -87.629645)
Guildford	(51.246194, -0.574250)
Hong Kong	(22.280144, 114.158341)
London	(51.514920, -0.090570)
New York	(40.748400, -73.985700)
Philly	(39.952364, -75.163616
Singapore	(1.280999, 103.845047)
Tokyo	(35.680886, 139.777483)
Boston	(42.358191, -71.060911)

Table 1. Central coordinates of each city graph within the SpaGBOL dataset.

Satellite Images: From each node's coordinate, we create a patch bounding box (configurable argument) with 50m side lengths at a zoom of 20, resulting in an approximate satellite image resolution of 0.2m/pixel. This is achieved as the images come from a combination of satellite and aerial systems, depending on the zoom level. The retrieved satellite image is north-aligned and resized to $[1000, 1000]$ and stored as RGB PNG files.

Runtime Data Processing

As shown in Table 1 within the main paper, the 10 city graphs are first separated into 9 training cities, and 1 test city, wholly unseen during training - Boston. Each of the 9 training graphs have the top-right ninth separated and edges removed for use as a validation set.

Training Operation: Depth-first walks are sampled randomly for each node within the training graphs. The satellite image of each node is selected, and the streetview image of each node is randomly selected from their set. These go

Model	Backbone	Dims ↓	Params (M) ↓	FLOPs (G) ↓	Inference Time (ms) ↓
CVM [4]	VGG16	4096	160.3	-	1228
CVFT [5]	VGG16	4096	26.8	-	369
DSM [6]	VGG16	4096	**14.5**	39.30	541
L2LTR [7]	HybridViT	768	195.9	46.77	170
GeoDTR+ [8]	ConvNeXt-T	768	24.7	11.25	161
SAIG-D [9]	SAIG-D	**384**	16.0	4.90	201
Sample4Geo [10]	ConvNeXt-T	768	28.6	**4.50**	**148**
SpaGBOL	ConvNeXt-T	768	56.0	17.82	430

Table 2. Model complexities comparison, with median query execution time on a single NVIDIA RTX 3090.

through the SpaGBOL network with streetview and satellite images going through the corresponding branches.

Evaluation Operation: At validation and testing stages, walks are exhaustively sampled from the graphs - with all satellite walks being embedded and stored in a KDTree. One walk per node is randomly sampled for querying the KDTree, being deemed correctly localised if the same target node is retrieved.

3. Model Characteristics

Execution time is crucial for online localisation in robotics applications. To further evaluate our proposed system against previous works we compare model complexities and query execution time. Table 2 displays various characteristics that impact the runtime of a model. From inferred embeddings, KDTree query time is then at best $O(\sqrt{D} + k)$ time, where k is the number of retrievals, and D the dimensionality.

References

[1] Geoff Boeing. Osmnx: A python package to work with graph-theoretic openstreetmap street networks. *The Journal of Open Source Software*, 2(12):215, Apr 2017. 1

[2] Streetview - pypi. 1

[3] Satellite imagery downloader. 1

[4] Sixing Hu, Mengdan Feng, Rang M. H. Nguyen, and Gim Hee Lee. Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7258–7267, 2018. 2

[5] Yujiao Shi, Xin Yu, Liu Liu, Tong Zhang, and Hongdong Li. Optimal feature transport for cross-view image geo-localization. *ArXiv*, 2019. 2

[6] Yujiao Shi, Xin Yu, Dylan Campbell, and Hongdong Li. Where am i looking at? joint location and orientation estimation by cross-view matching. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4063–4071, 2020. 2

[7] Hongji Yang, Xiufan Lu, and Ying J. Zhu. Cross-view geo-localization with layer-to-layer transformer. In *Neural Information Processing Systems*, 2021. 2

[8] Xiaohan Zhang, Xingyu Li, Waqas Sultani, Chen Chen, and Safwan Wshah. Geodtr+: Toward generic cross-view geolocalization via geometric disentanglement, 2023. 2

[9] Yingying Zhu, Hongji Yang, Yuxin Lu, and Qiang Huang. Simple, effective and general: A new backbone for cross-view image geo-localization, 2023. 2

[10] Fabian Deuser, Konrad Habel, and Norbert Oswald. Sample4geo: Hard negative sampling for cross-view geo-localisation, 2023. 2