

Highlights

OSMLoc: Single Image-Based Visual Localization in OpenStreetMap with Geometric and Semantic Guidances

Youqi Liao,Xieyuanli Chen,Shuhao Kang,Jianping Li,Zhen Dong,Hongchao Fan,Bisheng Yang

- A brain-inspired framework, OSMLoc, which jointly learns the geometry, semantics, and pose for image-to-OpenStreetMap (I2O) localization;
- A geometry-guided camera-to-BEV transformation module with depth prior from the foundation model;
- An auxiliary semantic alignment task to improve the scene understanding capability of the image model;
- A cross-area and cross-condition validation benchmark for extensive evaluation;

OSMLoc: Single Image-Based Visual Localization in OpenStreetMap with Geometric and Semantic Guidances

Youqi Liao^{a,*}, Xieyuanli Chen^{b,*}, Shuhao Kang^c, Jianping Li^{d,**}, Zhen Dong^a, Hongchao Fan^e and Bisheng Yang^a

^aWuhan University, Wuhan, 430079, China

^bNational Defense University of Techology, Changsha, 410003, China

^cTechnical University of Munich, Munich, D-80333, Germany

^dNanyang Technological University, Singapore, 639798, Singapore

^eNorwegian University of Science and Technology, Trondheim, NO-7491, Norwegian

ARTICLE INFO

Keywords:

Visual Localization

OpenStreetMap

Visual Foundational Model

ABSTRACT

OpenStreetMap (OSM), an online and versatile source of volunteered geographic information (VGI), is widely used for human self-localization by matching nearby visual observations with vectorized map data. However, due to the divergence in modalities and views, image-to-OSM (I2O) matching and localization remain challenging for robots, preventing the full utilization of VGI data in the unmanned ground vehicles and logistic industry. Inspired by the fact that the human brain relies on geometric and semantic understanding of sensory information for spatial localization tasks (Hermer and Spelke, 1994; Epstein and Vass, 2014), we propose the OSMLoc in this paper. OSMLoc is a brain-inspired single-image visual localization method with semantic and geometric guidance to improve accuracy, robustness, and generalization ability. First, we equip the OSMLoc with the visual foundational model to extract powerful image features. Second, a geometry-guided depth distribution adapter is proposed to bridge the monocular depth estimation and camera-to-BEV transform. Thirdly, the semantic embeddings from the OSM data are utilized as auxiliary guidance for image-to-OSM feature matching. To validate the proposed OSMLoc, we collect a worldwide cross-area and cross-condition (CC) benchmark for extensive evaluation. Experiments on the MGL dataset, CC validation benchmark, and KITTI dataset have demonstrated the superiority of our method. Code, pre-trained models, CC validation benchmark, and additional results are available on: <https://github.com/WHU-USI3DV/OSMLoc>.

1. Introduction

Map Construction and positioning are the basics of seamless navigation for logistic robots and unmanned ground vehicles (UGV). Most existing industrial solutions use the geo-referenced images (Arandjelovic et al., 2016; Cheng et al., 2018; Hausler et al., 2021; Zhu et al., 2023) or pre-built 3D point cloud maps (Yu et al., 2021; Chen et al., 2022; Shi et al., 2022; Zou et al., 2023) as the reference map. However, constructing geo-referenced images or point cloud maps is expensive and difficult to update. Since the beginning of the Internet era, volunteering geo-informatics (VGI) has emerged as a new paradigm that creates, assembles, and disseminates geographic data by crowd-sourcing users instead of specific experts (Wang et al., 2024). With over 10 million registered contributors (OSM_Foundation, 2024), the OpenStreetMap (OSM) provides detailed and up-to-date data about stationary objects throughout the world, including infrastructure and other aspects of the built environment, points of interest, land use and land cover classifications, and topography (Fan et al., 2014; Vargas-Munoz et al., 2020). With rich geographic and semantic information, humans can easily localize themselves by comparing the surrounding views with the OSM web map. However, the image-to-OSM (I2O) localization process remains challenging for robots due to differing modalities and perspectives, which limits the

full potential of VGI data in the logistics industry and hinder the development of navigation with the crowd-sourcing map.

Due to above mentioned difficulties, research on I2O visual localization is scarce. Early I2O localization approaches, such as Samano et al. (2020) localize along the pre-defined routes, while Zhou et al. (2021) integrates deep image features into the Monte Carlo framework to improve the representation ability of the observation model. OrientNet (Sarlin et al., 2023) is the first end-to-end single-image I2O localization approach, which lifts the front-view image features to the birds-eye-view (BEV) and estimates the pose by dense matching. MaplocNet (Wu et al., 2024) proposed a coarse-to-fine registration pipeline for I2O localization and introduced the semantic labels from the autonomous driving datasets as auxiliary supervision. Although existing methods have achieved impressive results, we highlight that two key challenges remain unsolved: 1) effective modular transformation of visual observations for cross-modality I2O matching; 2) explainable localization and strong generalization across diverse environments worldwide.

Inspired by how the human brain uses geometric and semantic understanding from sensory information and long-term spatial knowledge for visual localization task (Hermer and Spelke, 1994; Vass and Epstein, 2013; Epstein and Vass, 2014), this paper introduces the OSMLoc, a single image-based visual localization framework with semantic and geometric guidance. The core idea of OSMLoc is illustrated in Fig. 1(a). We inherit the fundamental model with rich prior

*Equal Contribution

**Corresponding author

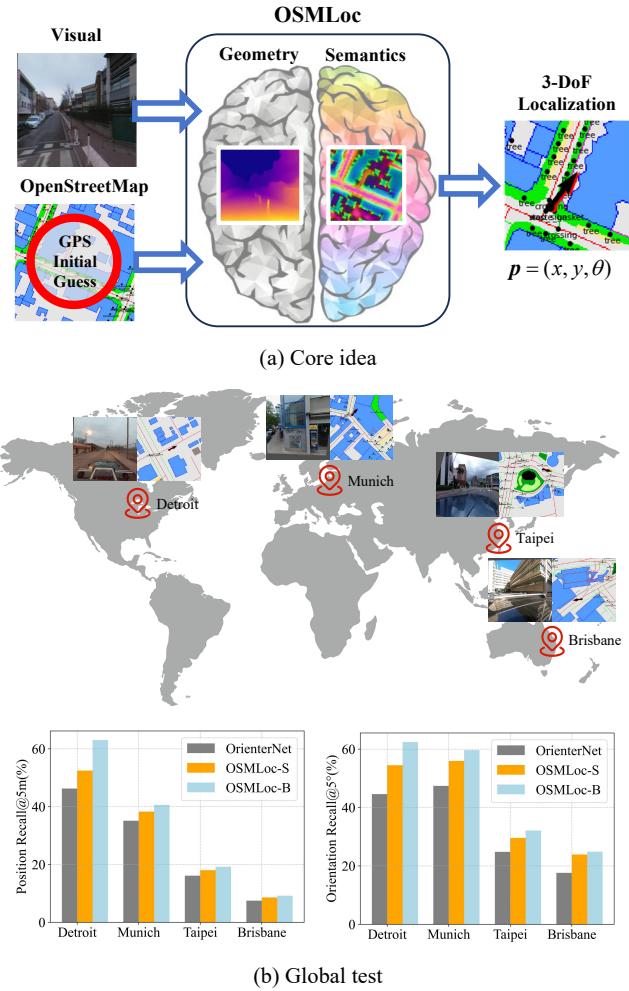


Figure 1: Image-to-OpenstreetMap localization aims to estimate the 3 degrees of freedom (3-DoF) camera pose. (a) shows the core idea of our method, which integrates geometry and semantic guidance into the framework. (b) shows the global evaluation results.

knowledge, specifically, the DINOv2 encoder (Oquab et al., 2023) of the Depth Anything model (Yang et al., 2024) into our framework to construct powerful image features. A compact depth distribution adapter is proposed to incorporate geometric information from Depth Anything into camera-to-BEV transformation. To our knowledge, this is the first attempt to integrate the foundation model into the I2O visual localization framework. The semantic consistency between visual and OSM data aids the model’s learning in completing cross-modular matching. Moreover, existing same-area validation sets exhibit similar styles, with consistent building and road patterns within neighborhood blocks of the training sets. This similarity arises because the data is typically collected by the same users, on similar devices, and within close timeframes. This uniformity may introduce bias into the validation results. Therefore, we collect a novel cross-area and cross-condition (CC) validation benchmark for evaluation in diverse environments, as shown in Fig. 1(b). The CC validation benchmark contains thousands of images

from Detroit, Munich, Taipei, and Brisbane with vast scenarios. Experiments on the Mapillary geo-localization (MGL) dataset, CC validation benchmark, and the KITTI dataset reveal that our method outperforms existing methods, particularly in unseen areas. Overall, the main contributions include:

1. We propose a brain-inspired framework, OSMLoc, which jointly learns the geometry, semantics, and pose for I2O visual localization. Given the challenges of localization across cross-view and cross-modality sources, we fully leverage geometric and semantic cues from the foundational model and OSM data to help the framework bridge the domain gap and understand the surroundings.
2. We propose a novel depth distribution adapter (DDA) to incorporate the depth prior to the camera-to-BEV transformation. Since accurate depth is essential for viewpoint transformation and monocular depth estimation is ambiguous, we utilize the DDA to distill the prior depth knowledge from the foundation model into our framework.
3. We introduce the auxiliary semantic alignment task to improve the scene understanding capability of the image model in a self-supervision manner. As the OSM data provides detailed semantic and geographic features of the areas, roads and other objects, we leverage the semantic consistency between the image and map as an additional constraint for the visual localization task.
4. We collect the cross-area and cross-condition (CC) validation benchmark with thousands of frames in Detroit, Munich, Taipei, and Brisbane, providing an extensive testbed for the I2O visual localization models.

2. Related works

Visual localization methods can be broadly categorized by data acquisition type into two groups: visual localization with expert data (Section 2.1) and VGI (Section 2.2). Expert data refers to data collected, processed and validated by professionals or specialized institutions with industrial-grade devices and rigorous methodologies, such as point cloud and aerial images. VGI is always created, assembled, and processed by individuals using consumer-grade devices and open-source platforms, such as street-view images captured by smartphones and manually curated OSM data.

2.1. Visual localization with expert data

The visual localization with point cloud has been studied for decades. Given a query image and pre-built point cloud map, the image-to-point cloud (I2P) visual localization task aims to estimate the transform parameters from the camera coordinate system to the point cloud coordinate system. Sattler et al. (2011) proposed a direct I2P matching framework by assigning point features to visual words and constructing the 2D-3D correspondences by linear search. Huitl et al.

(2012) proposed a novel image and point cloud dataset called TumIndoor to explore the I2P visual localization task in indoor environments. Due to the significant computational and storage burden of point cloud, Cheng et al. (2016) proposed a point cloud simplification framework based on the K-Cover theory to improve efficiency. Recent approaches (Li et al., 2020; Chen et al., 2022; Shi et al., 2022) focus on learning-based methods to improve accuracy and robustness. Some other approaches (Li and Lee, 2021; Kang et al., 2024) assume the coarse location of the image is accessible and aim to estimate the precise 6-DoF pose. Although point cloud contains detailed spatial information for localization, the extraction, processing, and storage are costly and require specialized sensors and software, limiting their applicability in consumer-grade scenarios.

With wide coverage and easy access, aerial image-based visual localization methods advance rapidly. Early image-to-aerial image (I2A) approaches (Lin et al., 2013; Tian et al., 2017) simplify the localization task by framing it as a retrieval problem. They divide the aerial image into non-overlapping patches and search for the most similar one in the feature space. VIGOR (Zhu et al., 2021) pointed out that one-to-one image retrieval is not suitable for practical scenarios. Consequently, the authors proposed a novel dataset and a coarse-to-fine pipeline that first retrieves the coarse pose and then regresses the fine pose. Shi and Li (2022) argued that directly regressing the camera pose is very difficult and proposed a multi-scale pose optimization method. The method projects the aerial image onto the ground image plane via geometric projection and searches for the optimal camera pose by minimizing the feature difference. Recently, SNAP (Sarlin et al., 2024) solves the ground image and aerial image visual localization, semantic segmentation, and mapping with a unified framework. As previous approaches rely on accurate pose labels for supervision, Li et al. (2024) proposed an unsupervised framework to fully leverage unlabeled data. Ye et al. (2024) proposed a coarse-to-fine I2A localization pipeline for oblique-view imagery captured by unmanned aerial vehicles. Although aerial images are more compact than 3D point cloud maps, they are still expensive to capture, generally not free, and heavy to store at high resolution. Moreover, aerial images of the same location vary due to different times, seasons, weather conditions, and lighting, making the long-term visual localization task more challenging.

2.2. Visual localization with VGI

Unlike expert data, VGI data is created, assembled, and shared by individuals. Visual localization with the geo-referenced ground images from VGI platforms, such as Mapillary (Warburg et al., 2020) or Google Maps (Google, 2024) as the primary databases, is a common research area. Traditionally, the image-to-ground images (I2I) visual localization approaches follow a two-step pipeline: 1) finding the coarse location first; 2) estimating the precise pose subsequently. In the first step, Arandjelovic et al. (2016); Li et al. (2023); Zhu et al. (2023) formulate the task as the visual

place recognition problem, which retrieves the top candidates first, and then re-ranks the top-k candidates to improve the performance further. In the second step, matching-based methods (Dusmanu et al., 2019; Sarlin et al., 2020) construct the pixel-to-pixel correspondences first and then estimate the pose by solving for the homography, whereas matching-free methods (Kendall et al., 2015; Liu et al., 2020; Tang et al., 2024) directly regress the 6-DoF transformation parameters with a pose regressor.

OpenStreetMap-based visual localization approaches are most related to our method. The pioneering approach OpenStreetSLAM (Floros et al., 2013) integrates visual odometry with map information to improve the robustness and accuracy of the vehicle’s pose estimation. It extracts the street graph from the map and uses it as the geo-reference for the observation model in the Monte Carlo localization (MCL) framework. Samano et al. (2020) embeds the images and map tiles into low-dimension feature vectors and searches for the most similar OSM tile in the pre-defined routes. Zhou et al. (2021) combines the deep image features with a sequential MCL framework. Yan et al. (2019); Cho et al. (2022); Lee and Ryu (2024) utilize the point cloud as visual observation instead of the image for localization with the OSM data. OrienterNet (Sarlin et al., 2023) is the first end-to-end single-image I2O visual localization framework that lifts the image feature into the BEV space and estimates the camera pose via feature matching. However, the monocular depth estimation in the view transformation is unreliable and hard to generalize to unseen environments. Maploc-Net (Wu et al., 2024) proposed a multi-task registration pipeline for simultaneously pose regression and semantic segmentation. The authors introduced the semantic labels from high-definition (HD) maps as auxiliary supervision. These labels are expertly annotated data and are not freely available. Unlike previous approaches, we propose a brain-inspired I2O visual localization framework that leverages geometric and semantic guidance from foundation models and OSM data, both are freely accessible and easily generalizable.

3. Our Approach: OSMLoc

Given a single image \mathcal{I} with noisy position prior $(\check{x}, \check{y}) \in \mathbb{R}^2$, our approach aims to query the precise position and orientation using the OSM map. As the OSM data mainly consists of 2D geospatial information, we simplify the typical 6-DoF pose estimation problem to the 3-DoF pose $\mathbf{p} = (x, y, \theta)$ consisting of the 2-D position $(x, y) \in \mathbb{R}^2$ and orientation angle $\theta \in (-\pi, \pi]$. We assume the gravity direction of the image is accessible via an inertial measurement unit or a gyroscope. The global coordinate system here is the topocentric coordinate system, with the x - y - z axes corresponding to *East-North-Up* directions.

Since our approach incorporates cross-view and cross-modality data, the image and OSM data require pre-processing before being fed into the framework. For the image \mathcal{I} , we rectify the roll and pitch angles to zero using the known gravity direction and ensure the principal axis is horizontal.

Areas	building, parking, playground, grass, park, forest, water
Ways	Fence, wall, hedge, kerb, building_outline, cycleway, path, road, busway, tree_row
Nodes	parking entrance, street lamp, junction, traffic signal, stop sign, give way sign, bus stop, stop area, crossing, gate, bollard, gas station, bicycle parking, charging station, shop, restaurant, bar, vending machine, pharmacy, tree, stone, ATM, toilets, water fountain, bench, waste basket, post box, artwork, recycling station, clock, fire hydrant, pole, street cabinet

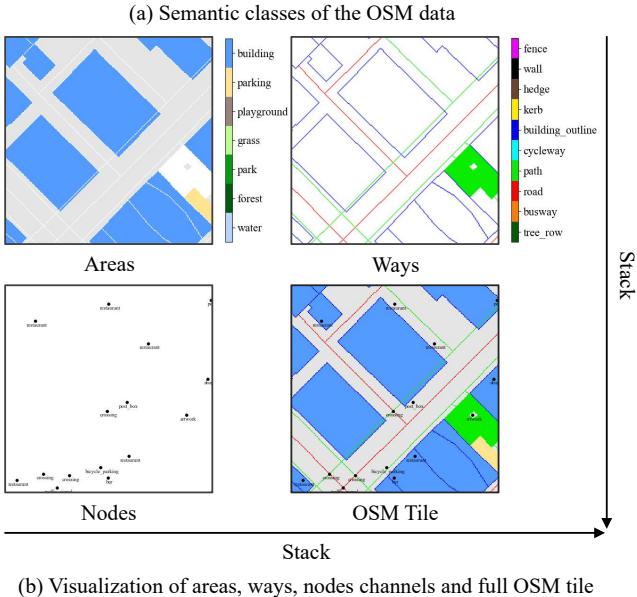


Figure 2: Illustration of the OSM data rasterization.

The OSM data is stored in structured metadata and represents the areas, ways and nodes with the region of interest (ROI), as well as polylines and point of interest (POI). We categories the elements as in Fig. 2(a) and project the OSM data onto the x - y plane of the global coordinate system for rasterization. As shown in Fig. 2(b), we rasterize the areas, ways and nodes into a 3-channel grid map with a fixed sampling distance $\Delta s = 50$ cm/pixel, while preserving the semantic and geographic information. During training and evaluation, we crop map tile \mathcal{M} of the rasterized OSM data centered around the location prior as input.

The workflow of our method is illustrated in Fig. 3. Our OSMLoc first embeds the image I and OSM map tile \mathcal{M} into high-dimensional feature space (Sec. 3.1), and then the depth distribution adapter distills the geometric prior from foundation model into depth predictions, while the camera-to-BEV transformation module (Sec. 3.2) lifts the image features to birds-eye-view. Lastly, the neural localization (Sec. 3.3) module estimates pose by dense feature matching. The semantic alignment is employed as an auxiliary task to improve the scene understanding ability of the image model. Pose classification loss \mathcal{L}_{pose} , depth distribution loss \mathcal{L}_{depth}

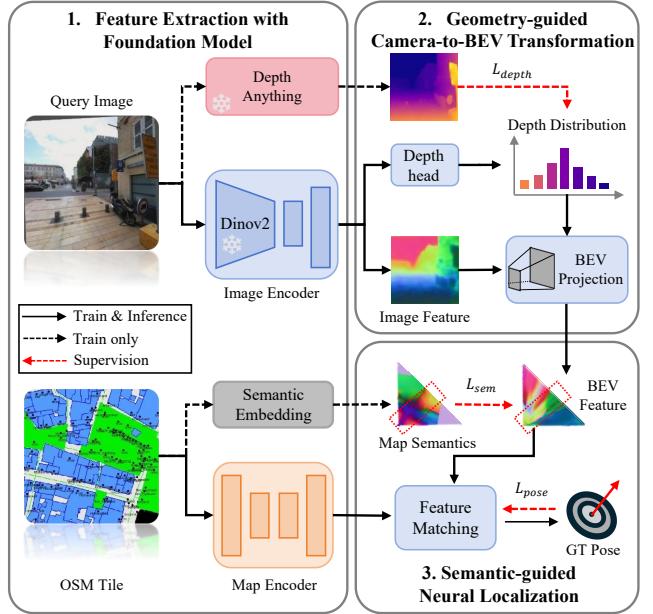


Figure 3: Workflow of the OSMLoc.

and semantic alignment loss \mathcal{L}_{sem} are leveraged to supervise the network jointly.

3.1. Feature extraction with foundation model

For the image model, we utilize the DINOv2 (Oquab et al., 2023) of Depth Anything (Yang et al., 2024) as the image encoder. DINOv2 (Oquab et al., 2023) trained the vision transformer (ViT) (Dosovitskiy, 2020) based model over 142 M images for general vision perception tasks. Depth Anything (Yang et al., 2024) fine-tuned the pre-trained DINOv2 model on over 1.5 M labeled images and 62 M unlabeled images for the depth estimation task. As the Depth Anything model shows strong zero-shot relative depth estimation and semantic capability in various scenes, which is advantageous for the view transformation and neural localization, we adopt its DINOv2 model as our image encoder and freeze the pre-trained weights during training. Since Anyloc (Keetha et al., 2023) demonstrated that features from multiple stages of foundational models are beneficial for visual localization, we extract multi-stage features from the image encoder and feed them into a learnable feature pyramid network (FPN) decoder for progressive fusion. Finally, the output image feature $F_{img} \in \mathbb{R}^{U \times V \times C}$ is bilinearly upsampled to the size $U \times V$ with C channel.

For the map model, we use a multi-layer perception (MLP) to embed the input 3-channel rasterized map into a high-dimensional semantic embedding $M_{sem} \in \mathbb{R}^{H \times W \times (3 \times C_{sem})}$, where H and W are the height and width of the map. Each semantic class of areas, roads and nodes is mapped to a unique semantic vector with C_{sem} channels, respectively. Besides, a VGG-16 model is employed as the map encoder, and the decoder is a compact FPN. Similar to the image model, we extract multi-stage features from the encoder

and feed them into the FPN to extract the final map feature $\mathbf{F}_{map} \in \mathbb{R}^{H \times W \times C}$.

3.2. Geometry-guided camera-to-BEV transformation

With the extracted image feature $\mathbf{F}_{img} \in \mathbb{R}^{U \times V \times C}$, we infer a BEV feature map $\mathbf{F}_{bev} \in \mathbb{R}^{L \times D \times C}$ in a $L \times D$ grid map on the x - y plane of the global coordinate system. Following the BEV perception methods (Philon and Fidler, 2020; Liu et al., 2023), for each pixel (u, v) on the image, we sample D discrete points $\{(u, d_i, v) \in \mathbb{R}^3 | d_i = d_0 + i\Delta, i \in \{1, 2, \dots, D\}\}$ among the camera ray and predict the associated depth distribution $\alpha_{u,v} \in \mathbb{R}^D$. Then we scatter the image feature to the points with the corresponding probability, which generates the feature point cloud $\mathbf{F}_{pc} \in \mathbb{R}^{UDV \times C}$ of size UDV . For each point (u, d_i, v) , the associated feature $\mathbf{F}_{pc}(u, d_i, v) \in \mathbb{R}^C$ is defined as the rescaled feature vector of pixel (u, v) :

$$\mathbf{F}_{pc}(u, d_i, v) = \alpha_{u,d_i,v} \mathbf{F}_{img}(u, v). \quad (1)$$

Since the image is gravity-aligned, each column corresponds to a triangular "slice" in the frustum (Lentsch et al., 2023), and the associated feature points should lie within the same rectangular region extending from the camera in the BEV polar coordinate system. Therefore, we assemble the feature point cloud of each column into a polar strip on the plane, resulting in a $D \times V$ polar feature map \mathbf{F}_{bev} . For each polar grid (d, v) , the feature projection could be formulated as:

$$\mathbf{F}_{bev}(d, v) = \sum_{u=1}^U \alpha_{u,d,v} \mathbf{F}_{pc}(u, d, v), \quad (2)$$

where $\mathbf{F}_{bev} \in \mathbb{R}^{D \times V \times C}$ is the BEV feature map in the polar coordinate system of the plane. Then we remap the BEV features from the polar coordinate system to a Cartesian grid map of size $D \times L$ by linear interpolation.

Depth distribution adapter: Although simple, unsupervised monocular depth estimation presents significant challenges. While the depth value might be available from other sensors (e.g., LiDAR or RGB-D camera) for supervision, it is a non-trivial assumption and hard to generalize to novel devices and unseen environments. Furthermore, for each pixel, converting the depth value $d \in (0, +\infty)$ into the discrete depth distribution $\alpha \in \mathbb{R}^D$ is also an ill-posed problem.

Fortunately, Depth Anything (Yang et al., 2024) offers a zero-shot relative depth estimation model, where the output normalized disparity $\bar{t} \in [0, 1]$ provides powerful prior information about the relative distance relationship for view transformation. The normalized disparity \bar{t} is a scale- and shift-invariant, numerically stable version of the depth d . During supervised training, the authors first transform the depth value d into the disparity t by $t = 1/d$ and then normalize it to the \bar{t} for each image:

$$\bar{t} = \frac{t - \min(t)}{\max(t) - \min(t)}, \quad (3)$$

where $\max(t)$ and $\min(t)$ are the maximal and minimal disparity values within the image. However, both $\max(t)$ and $\min(t)$ are unavailable during inference, and the normalized disparity \bar{t} cannot be reversed to the depth d or the depth distribution α for the view transformation.

Considering the above issue, we design the depth distribution adapter (DDA) to distill the prior knowledge from the Depth Anything (Yang et al., 2024) model into the camera-to-BEV transformation, as shown in Fig. 4. With the predicted depth distribution $\alpha_{u,v} \in \mathbb{R}^D$ for each pixel (u, v) , the depth value $d_{u,v}$ is formatted as the weighted average value of the depth distribution:

$$d_{u,v} = \sum_{i=1}^D \alpha_{u,d_i,v} d_i, \quad s.t. \sum_{i=1}^D \alpha_{u,d_i,v} = 1. \quad (4)$$

Then we transform the depth estimation $d_{u,v}$ to normalized disparity $\bar{t}_{u,v}$ according to Eq. (3). With the pseudo ground truth (GT) disparity value $\hat{t}_{u,v}$ from the decoder of the Depth Anything model, the depth distribution loss \mathcal{L}_{depth} is defined as the mean absolute error:

$$\mathcal{L}_{depth} = \frac{1}{UV} \sum_{u=1}^U \sum_{v=1}^V |\bar{t}_{u,v} - \hat{t}_{u,v}|. \quad (5)$$

By aligning the disparity predictions with Depth Anything, the DDA module effectively incorporates geometric prior from the foundation model into the framework, enabling it to understand the topology of surrounding objects. During inference, we discard the disparity supervision and directly utilize the depth distribution α for the camera-to-BEV transform.

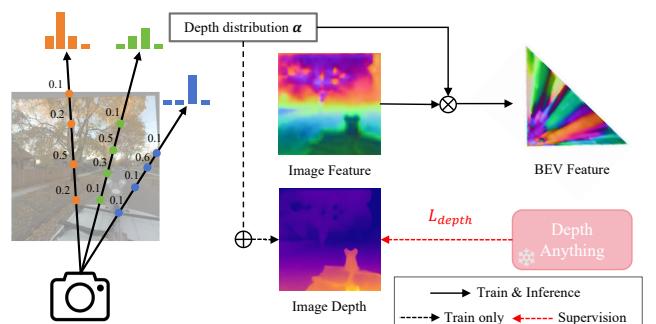


Figure 4: Illustration of the depth distribution adapter. For each pixel, we predict the depth distribution $\alpha \in \mathbb{R}^D$ and scatter the image feature \mathbf{F}_{img} to the BEV feature \mathbf{F}_{bev} by the corresponding depth distribution α . During training, the depth distribution prediction is transferred to the depth map and supervised with the Depth Anything.

3.3. Semantic-guided neural localization

Since OSM data provides semantic and geographic information about stationary objects, the key to successful localization lies in constructing the BEV feature \mathbf{F}_{bev} with consistent semantics and topology to those of the OSM data. Explicit feature registration at the semantic level also

aligns with how humans localize themselves (Epstein and Vass, 2014). However, previous I2O visual localization approaches largely overlooked the semantic consistency of images and OSM data. OrienterNet (Sarlin et al., 2023) exploited the network for pose estimation in an end-to-end way. However, pose supervision only makes it challenging for the model to fully understand the surroundings. Concurrent approach MaplocNet (Wu et al., 2024) pointed out that the semantic labels from the HD maps could improve the comprehension ability of the image model and substantially enhance visual localization accuracy. However, these semantic labels are not freely accessible and limit the method's generalizability. To address the challenge, we introduce an auxiliary semantic alignment task to improve the scene understanding capability of the image model. The semantic alignment task fully utilizes the semantic embeddings \mathcal{M}_{sem} as guidance and is generally free. During the training phase, for each grid (l, d_i) on the BEV feature $\mathbf{F}_{bev} \in \mathbb{R}^{L \times D \times C}$, we find the corresponding area on the map tile \mathcal{M} and the associated semantic embedding vector with the GT pose $\hat{\mathbf{p}}$. Then, we directly align the BEV features \mathbf{F}_{bev} with the corresponding semantic embeddings of the OSM data. The semantic alignment loss \mathcal{L}_{sem} is defined as:

$$\mathcal{L}_{sem} = \frac{1}{LD} \sum_{l=1}^L \sum_{i=1}^D \|C_{1 \times 1}(\mathbf{F}_{bev}[l, d_i]) - \mathbf{M}_{sem}[\hat{\Gamma}(l, d_i)]\|_2. \quad (6)$$

$\hat{\Gamma}$ is the transformation matrix of GT pose $\hat{\mathbf{p}}$ and $C_{1 \times 1}$ is a 1×1 convolution layer to adjust the channel dimension of the BEV features.

With the semantic-rich BEV features \mathbf{F}_{bev} and map features \mathbf{F}_{map} , the neural localization is responsible for estimating the 3-DoF pose $\mathbf{p} = (x, y, \theta)$ of the image. The most straightforward strategy is, fusing the BEV features and map features and estimating the pose with a small regressor. Unfortunately, in our pilot studies, the network failed to converge with the regression design after trial and error, which contradicts the observation in MaplocNet (Wu et al., 2024). We speculate that this is due to the limited field of view (FoV) of a single image, which results in the BEV feature \mathbf{F}_{bev} containing partial information. Therefore, directly learning the pose vector is challenging. To simplify the problem, we discretely sample the continuous 3-D pose space at each map grid and K rotations with regular intervals, resulting in a $H \times W \times K$ volume of pose candidates in the 3-DoF pose space. For each pose candidate $\mathbf{p}_{h,w,k} = (x_h, y_w, \theta_k)$, we project the BEV feature \mathbf{F}_{bev} onto the map plane and calculate the matching score in the feature space :

$$\mathbf{S}_{h,w,k} = \frac{1}{LD} \sum_{l=1}^L \sum_{i=1}^D \mathbf{F}_{map}(\Gamma_{h,w,k}(l, d_i)) \odot \mathbf{F}_{bev}(l, d_i), \quad (7)$$

where $\Gamma_{h,w,k}$ is the transformation matrix for pose candidate $\mathbf{p}_{h,w,k}$, and \odot denotes the inner product operation. Eq. 7 benefits from an efficient implementation by performing a

single convolution as a batched multiplication in the Fourier domain (Barsan et al., 2018). Finally, the scoring volume \mathbf{S} is normalized to the probability volume \mathbf{P} with the softmax operation.

The pose classification loss \mathcal{L}_{pose} is defined as a binary cross-entropy loss to maximize the probability of the GT pose $\hat{\mathbf{p}}$:

$$\mathcal{L}_{pose} = - \sum_{h=1}^H \sum_{w=1}^W \sum_{k=1}^K \hat{P}_{h,w,k} \log(P_{h,w,k}) = -\log P_{\hat{x}, \hat{y}, \hat{\theta}}, \quad (8)$$

where the \hat{P} is the probability volume of the GT pose, which is concentrated at a single point $\hat{\mathbf{p}} = (\hat{x}, \hat{y}, \hat{\theta})$ in the pose space.

Overall, the joint loss \mathcal{L} is defined as a weighted combination of pose classification loss \mathcal{L}_{pose} , depth distribution loss \mathcal{L}_{depth} and semantic alignment loss \mathcal{L}_{sem} :

$$\mathcal{L} = \lambda_1 \mathcal{L}_{pose} + \lambda_2 \mathcal{L}_{depth} + \lambda_3 \mathcal{L}_{sem}, \quad (9)$$

where the $\lambda_1, \lambda_2, \lambda_3$ are hyper-parameters to control the weights of each component.

During the inference, we estimate the pose \mathbf{p}^* as the one corresponding to the peak in the probability volume \mathbf{P} :

$$\mathbf{p}^* = \operatorname{argmax}_{\mathbf{p}} P(\mathbf{p} | \mathcal{I}, \mathcal{M}) \quad (10)$$

4. Cross-area and cross-condition validation benchmark

OrienterNet (Sarlin et al., 2023) has constructed the Mapillary geo-localization (MGL) dataset for the I2O visual localization task, containing 826 K images for training and 2 K images for validation. The training-evaluation split follows the "same-area" protocol, where each location's images are disjointed into training and validation sets. Besides, the distribution of cameras, motions, viewing conditions, and visual features of the validation set is identical to the training set. Such **same-area** and **same-condition** setup may work well for localization within a small, local area over a short time span, it is not appropriate for assessing the performance of I2O localization methods across different countries with VGI data collected at various conditions.

To address this concern, we propose a novel validation benchmark for **cross-area** and **cross-condition** validation, called the **CC** validation benchmark in the following. Table. 1 lists details of the CC validation benchmark, which contains 7164 images from 4 cities on the Mapillary platform (Mapillary, 2024). These images were collected with diverse devices, by different users, with varying motion patterns and under various external conditions, as shown in Fig. 5 and Fig. 6. They have never been seen before and are located hundreds of miles away from the nearest area in the MGL dataset. Each image is captured by cameras handheld or mounted on helmets, UGVs, or vehicles. For each 360° panorama, we split it into 4 × 90° FoV perspective images with a random offset and resample them to a size of 512×512 pixels. We query the OSM data for the corresponding area

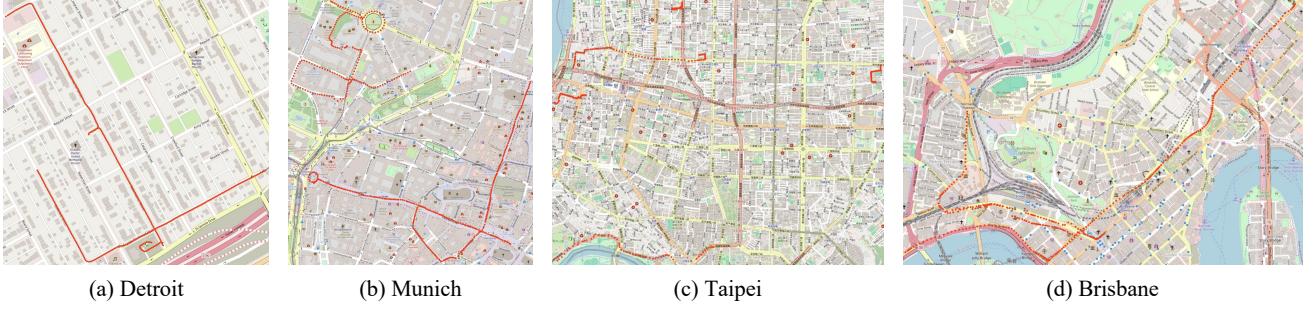


Figure 5: OSM data coverage and image trajectories in the CC validation benchmark. The red points represent the locations of the images.

from the official website (OSM, 2024) and the data pre-processing is shown in Fig. 2. The validation scenes in the CC validation benchmark cover a broad range of urban scenes, including downtown, suburban, overpass regions, and pedestrian streets. We believe the diverse set of scenes, captured by various cameras, users, platforms, and under different external conditions, makes the CC benchmark more challenging and suitable for comprehensive evaluation.

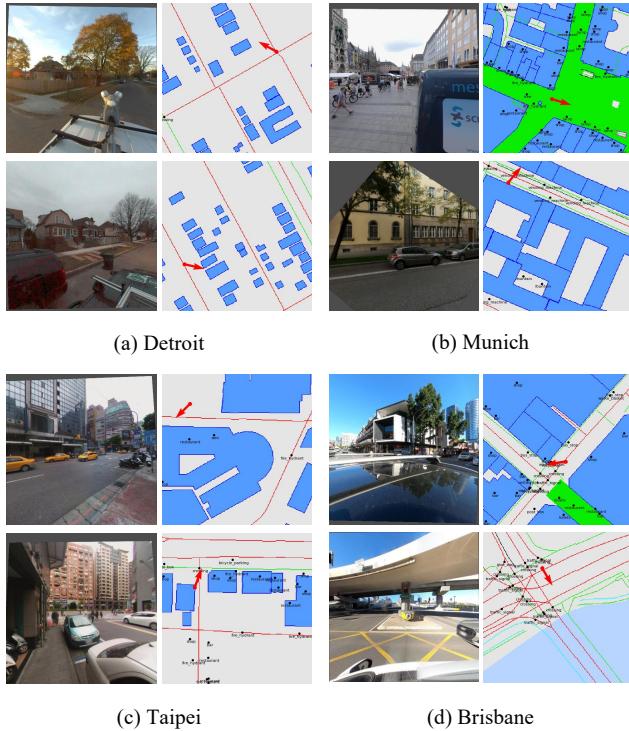


Figure 6: Query image and map tile pairs from the CC validation benchmark. Red arrow ↑ represents the GT pose in the map tile.

5. Experiment

5.1. Experimental setup

We evaluate the performance of our OSMLoc on three datasets: the MGL dataset, the CC validation benchmark, and the KITTI (Geiger et al., 2012) dataset. Our model is

Table 1

Details of CC validation benchmark.

City	Image	Camera	Platform
Detroit	1676	Trimble mx50 Gray Search Ladybug	Vehicle
Munich	2156	Trimble mx50 Labpano Pilot Era	Handheld UGV
Taipei	1872	LG-R105 RICOH THETA V	Handheld Vehicle
Brisbane	1460	Garmin VIRB 360 GoPro MAX	Vehicle

trained on the MGL dataset and directly validated on the CC validation benchmark and KITTI dataset without fine-tuning.

- MGL dataset: consists of 2580 sequences with approximately 828 K images collected in 13 cities from the Mapillary platform, which exposes the camera calibration, noise GPS measurement, and the 3-DoF pose in a global reference frame obtained by the fusion of motion pattern and GPS. All the sequences were recorded after 2017 with cameras known for high-quality reconstructions. Images of each city are split into training and validation sets, resulting in 826 K training and 2 K validation images.
- CC validation benchmark: consists of 14 long sequences with 7164 images from Detroit, Munich, Taipei, and Brisbane captured between 2017 to 2024. The regions have never been seen in the MGL dataset. More details can be found in Sec. 4.
- KITTI dataset: provides the stereo images captured by a moving vehicle along different trajectories, primarily used for autonomous driving. We use the KITTI dataset to evaluate the generalization capability of the proposed method across various image-capturing platforms. The GT poses were collected from the integrated navigation system. Shi and Li (2022) split the KITTI dataset into the *Train*, *Test1* and *Test2* sets for I2A visual localization task. The *Test2* set was collected in a different area, with no overlap of the MGL dataset or the KITTI *Train* set. Following the OrienterNet (Sarlin et al., 2023), we select the *Test2*

as the validation set, assuming that an accurate initial pose sampled within $\pm 20\text{m}$ and $\pm 10^\circ$ is available. Only the pose candidates that fall within the range are considered during inference.

Baseline methods: Due to the limited focus on the single-image I2O visual localization, OrieneterNet (Sarlin et al., 2023) is currently the only open-sourced comparable method in the field. We introduce variants of I2A visual localization methods (Shi and Li, 2022; Xia et al., 2022) as baselines for further comparison.

- OrieneterNet (Sarlin et al., 2023): is the pioneering end-to-end I2O visual localization method for single image. We use the officially released code for implementation and retrain the model with the same settings, except for setting the batch size to 4.
- Retrieval (Xia et al., 2022): replaces the BEV inference and matching with a correlation between the neural map and a global image embedding. As the original implementation predicts a 2-DoF position and ignores the orientation, we predict the orientation by considering four neural maps for the *North-South-East-West* directions. This formulation also regresses a probability volume, similar to OrieneterNet.
- Refinement (Shi and Li, 2022): updates an initial pose by warping a satellite view to the image assuming that the scene is planar, at a fixed height, and gravity-aligned. We replace the satellite view with a rasterized OSM tile. This formulation requires an initial orientation (both during training and testing), which we sample within 45° of the ground truth orientation angle.

Evaluation metrics: For the MGL dataset and the CC validation benchmark, we calculate the position recall (PR), orientation recall (OR), absolute position error (APE) and absolute orientation error (AOE) as the evaluation metrics. With the predicted pose $p_i = (x_i, y_i, \theta_i)$ and GT pose $\hat{p}_i = (\hat{x}_i, \hat{y}_i, \hat{\theta}_i)$ of frame i , the PR and OR are formulated as :

$$\begin{aligned} \text{PR} &= \frac{1}{M} \sum_{i=1}^M \mathbb{1}(\|(x_i, y_i) - (\hat{x}_i, \hat{y}_i)\|_2 < \sigma_p), \\ \text{OR} &= \frac{1}{M} \sum_{i=1}^M \mathbb{1}(|\theta_i - \hat{\theta}_i| < \sigma_o), \end{aligned} \quad (11)$$

where M is the number of test frames and the threshold pairs $\sigma = (\sigma_p, \sigma_o)$ are the maximal tolerance of the localization error. The APE and AOE are the mean absolute error between the predicted poses and GT poses:

$$\begin{aligned} \text{APE} &= \frac{1}{M} \sum_{i=1}^M \|(x_i, y_i) - (\hat{x}_i, \hat{y}_i)\|_2, \\ \text{AOE} &= \frac{1}{M} \sum_{i=1}^M |\theta_i - \hat{\theta}_i|. \end{aligned} \quad (12)$$

For the KITTI dataset, since position recall is closely related to the vehicle’s motion direction in autonomous driving scenes (Shi and Li, 2022), we separate the position recalls into two components: the perpendicular direction recall (latitudinal, LatR) and the parallel direction recall (longitudinal, LonR) to the viewing axis.

Implementation details: For the image, we set U and V to half of the original image size and $D = 64$, $d_0 = 0$, $\Delta = 0.5\text{m}$ and $L = 129$ during the camera-to-BEV transformation. For the OSM data, we render map tiles of size $H \times W = 128\text{m} \times 128\text{m}$ centered around points randomly sampled within 32m of the ground truth pose. The orientation sampling number K is set to 64 during training and 256 during inference. The GT pose \hat{p} , pseudo disparity label \hat{t} and the semantic embedding \mathcal{M}_{sem} are jointly used to supervise the network. We trained the network for 500K steps with a batch size of 4, using SGD to optimize the network, and the initial learning rate is set to 1e-4. The hyperparameters for the loss weight are set to $\lambda_1 = 1$, $\lambda_2 = 10$, $\lambda_3 = 20$. The threshold pairs $\sigma = (\sigma_p, \sigma_o)$ are set to $(1\text{m}, 1^\circ)$, $(3\text{m}, 3^\circ)$ and $(5\text{m}, 5^\circ)$ during evaluation. All the experiments are conducted on a computer with an Intel i9-13900K CPU and an NVIDIA RTX 4090 GPU. By adjusting the layers and channels of the image model, we design two variants of our method: the small and base versions (denoted as OSMLoc-S and OSMLoc-B, respectively). More configurations and implementation details are available in our open-source code¹.

5.2. Localization results

MGL dataset: Table. 2 shows the single-image visual localization results on the MGL dataset. Notably, our OSMLoc significantly outperforms all baseline methods by a large margin across all position and orientation recall thresholds. Specifically, OSMLoc-S reduces the APE and the AOE by approximately $0.91\text{m}/2.62^\circ$ and improves the $5\text{m}/5^\circ$ recall by $3.30\%/4.82\%$ over the SOTA baseline method, OrieneterNet (Sarlin et al., 2023); OSMLoc-B achieves the best performance, with a $1.96\text{m}/5.80^\circ$ reduction in APE and AOE and a $5.39\%/7.70\%$ improvement in the $5\text{m}/5^\circ$ recall. With comparable inference speed, the improvement mainly results from incorporating geometric and semantic guidance into the I2O localization framework rather than enlarging the model. Furthermore, Fig. 7 shows the recall curves for top candidates at different thresholds, where OSMLoc-B achieves the highest recall across various settings. These quantitative results demonstrate that our method significantly enhances prediction accuracy and robustness, as evidenced by consistent gains in position and orientation recall across different thresholds and top candidates.

CC validation benchmark: We evaluate our method and the baseline method OrieneterNet, on the CC validation benchmark and report the single-image localization results in Table. 3. Although the CC validation benchmark is more challenging than the same-area and same-condition validation set of the MGL dataset, our method consistently

¹<https://github.com/WHU-USI3DV/OSMLoc>

Table 2

Visual localization results on the MGL dataset. ↑ means higher is better, ↓ means lower is better. The localization results of "Retrieval" and "Refinement" are taken from OrienterNet (Sarlin et al., 2023). We use **bold** font to indicate the best results and underline to indicate the second-best results.

Approach	PR @Xm↑				OR @X°↑			APE(m)↓	AOE(°)↓	FPS↑
	@1m	@3m	@5m	@1°	@3°	@5°				
Retrieval	2.02	15.21	24.21	4.50	18.61	32.48	-	-	-	-
Refinement	8.09	26.02	35.31	14.92	36.87	45.19	-	-	-	-
OrienterNet	10.69	42.12	54.01	19.22	48.56	63.59	12.00	28.71	<u>5.5</u>	-
OSMLoc-S	14.04	44.00	57.31	21.84	53.38	68.41	11.09	26.09	6.2	-
OSMLoc-B	15.30	46.05	59.40	<u>21.79</u>	56.57	71.29	10.04	22.91	5.0	-

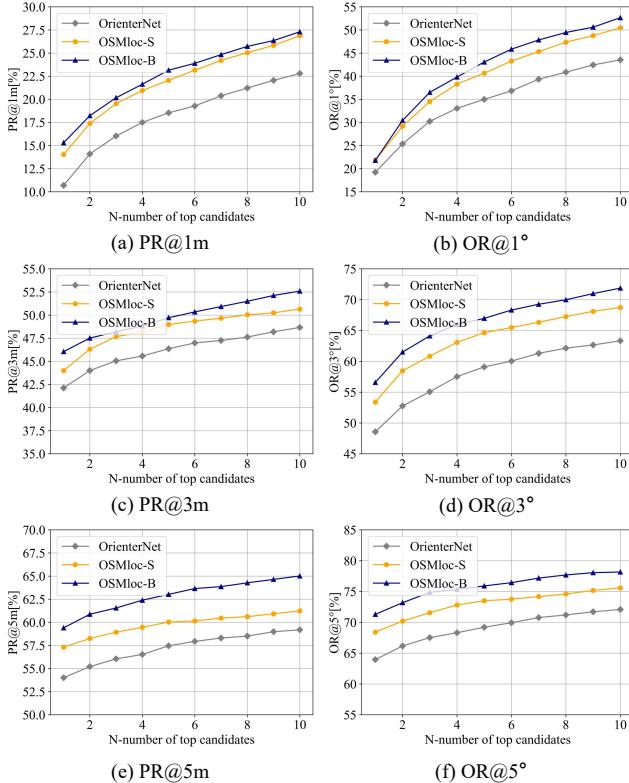


Figure 7: Position and orientation recall curves of the top candidates on the MGL dataset. The gray, orange, and navy curves are for OrienterNet, OSMLoc-S and OSMLoc-B respectively.

outperforms the baseline across all four cities, demonstrating a clear advantage in various scenes. Specifically, our OSMLoc-B achieves 16.77%/17.84% and 5.47%/12.25% improvement of 5m/5° recall in Detroit and Munich respectively, demonstrating the superiority of our method. Since the test areas of Taipei and Brisbane are situated in central urban districts with numerous skyscrapers and overpasses, these scenes differ significantly from the training set and offer limited useful information for localization. We believe that these factors contribute to the relatively lower localization success rate in these regions. Nevertheless, our OSMLoc still achieves significantly higher localization accuracy compared to the baseline. Fig. 8 presents the qualitative results of OSMLoc-S on the CC validation benchmark.

Table 3

Visual localization results on the CC validation benchmark. ↑ means higher is better. We use **bold** font to indicate the best results.

City	Method	PR @Xm↑			OR @X°↑		
		@1m	@3m	@5m	@1°	@3°	@5°
Detroit	OrienterNet	7.64	33.11	46.24	13.66	32.10	44.57
	OSMLoc-S	11.99	43.32	52.45	16.77	39.86	54.47
	OSMLoc-B	13.37	51.79	63.01	19.75	48.57	62.41
Munich	OrienterNet	2.68	17.90	35.11	12.48	32.75	47.40
	OSMLoc-S	3.20	19.53	38.27	15.07	39.19	55.94
	OSMLoc-B	4.36	22.03	40.58	15.35	41.79	59.65
Taipei	OrienterNet	1.23	8.17	16.13	5.88	16.40	24.79
	OSMLoc-S	1.07	8.39	18.00	7.00	20.35	29.59
	OSMLoc-B	1.44	9.72	19.18	7.32	21.31	32.10
Brisbane	OrienterNet	0.21	2.60	7.47	4.66	9.73	17.60
	OSMLoc-S	0.41	2.81	8.56	5.55	15.34	23.90
	OSMLoc-B	0.27	2.88	9.18	5.00	15.34	24.86

KITTI dataset: Extensive experiments are conducted on the KITTI dataset to demonstrate the generalizability of our method in self-driving scenarios. The quantitative results are shown in Table. 4. Without fine-tuning, our OSMLoc outperforms baseline methods trained on the KITTI dataset by a large margin, demonstrating its strong generalization capability and robustness. Furthermore, compared with the methods (Shi and Li, 2022; Sarlin et al., 2023) trained on the MGL dataset, our OSMLoc achieves much higher position and orientation recall. These quantitative results highlight that our method excels in accuracy, robustness, and generalization, making it suitable for versatile downstream tasks. Fig. 9 deploys additional qualitative results, where our method produces cleaner likelihood maps and more accurate camera pose predictions.

5.3. Ablation studies and Analysis

In this section, we analyze the impact of the foundation model in the image encoder, as well as the geometric and semantic guidance, on the overall framework. Ablation experiments are conducted on the MGL dataset to assess the effectiveness of each module. We also present some failure cases to explore the limitations of our approach.

Firstly, we conduct ablation studies to investigate the influence of the foundational model-based image encoder. As shown in Table. 5, the foundation model boosts the localization performance with its rich prior knowledge. Additionally, utilizing the larger DINOv2-B leads to continuous

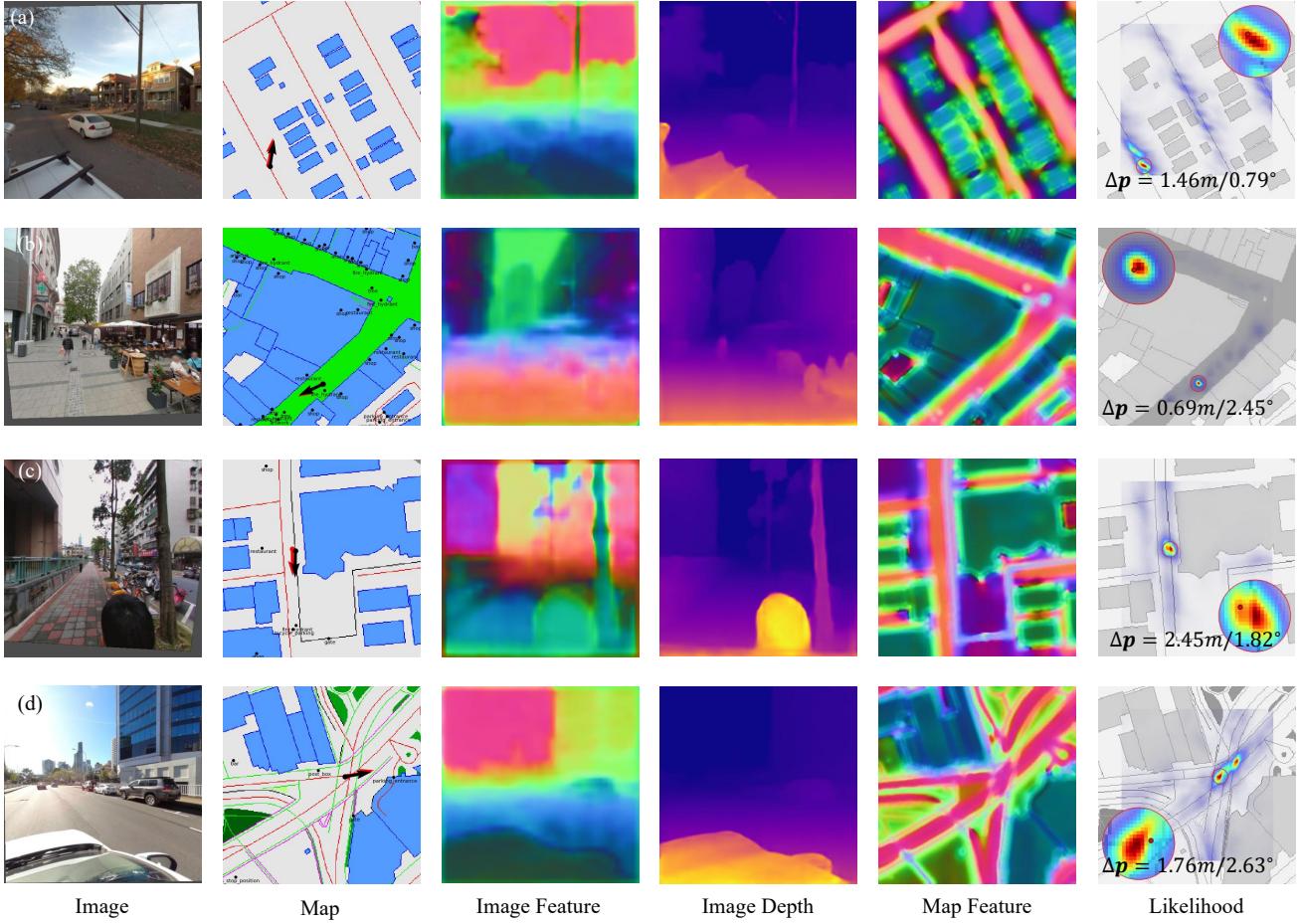


Figure 8: Qualitative results of OSMLoc-S on the CC validation benchmark. (a) Detroit. (b) Munich. (c) Taipei. (d) Brisbane. Black arrow ↑ represents the predicted pose and red arrow ↑ represents the GT pose in the map.

Table 4

Quantitative results on the KITTI dataset. We use **bold** font to indicate the best results and underline to indicate the second-best results..

Map	Method	Training dataset	LatR @Xm↑			LonR @Xm↑			OR @X°↑		
			@1m	@3m	@5m	@1m	@3m	@5m	@1°	@3°	@5°
Satellite	DSM	KITTI	10.77	31.37	48.24	3.87	11.73	19.50	3.53	14.09	23.95
	VIGOR		17.38	48.20	70.79	4.07	12.52	20.14	-	-	-
	Refinement		27.82	59.79	72.89	5.75	16.36	26.48	18.42	49.72	71.00
OSM	Retrieval	MGL	37.47	66.24	72.89	5.94	16.88	26.97	2.97	12.32	23.27
	Refinement		50.83	78.10	82.22	17.75	40.32	52.40	31.03	66.76	76.07
	OrienterNet		62.03	90.89	95.73	19.99	51.60	65.78	29.93	72.81	87.79
	OSMLoc-S		65.75	<u>93.79</u>	<u>96.74</u>	<u>24.50</u>	<u>59.53</u>	<u>72.67</u>	<u>33.41</u>	<u>75.67</u>	<u>91.67</u>
	OSMLoc-B		66.71	94.26	97.04	27.14	62.09	73.80	37.43	79.69	92.61

improvement in localization accuracy, but it also increases the computational complexity and reduces the efficiency. Table. 6 shows the ablation study results on the depth guidance and semantic guidance. In variant [A], the framework is supervised by the pose label \hat{p} only. Variant [B] utilizes the geometric guidance from the Depth Anything and introduces the depth loss \mathcal{L}_{depth} into the framework. The localization results show that geometric guidance is beneficial for accurate localization. Variant [C] utilizes the semantic embeddings M_{sem} from the OSM data and introduces the semantic alignment loss \mathcal{L}_{sem} , which also boosts the PR performance.

Finally, variant [D] demonstrates that jointly learning the pose, geometric and semantic achieves the best localization performance.

Failure cases: The localization accuracy of our OSMLoc heavily depends on the quality, quantity and distinctiveness of the surrounding objects in the image. When the camera's FoV is obstructed by skyscrapers or the surrounding environment is monotonous, our method may struggle to accurately estimate the pose. Fig. 10 shows some failure cases under different conditions. In the first and second cases, useful information is limited due to the buildings

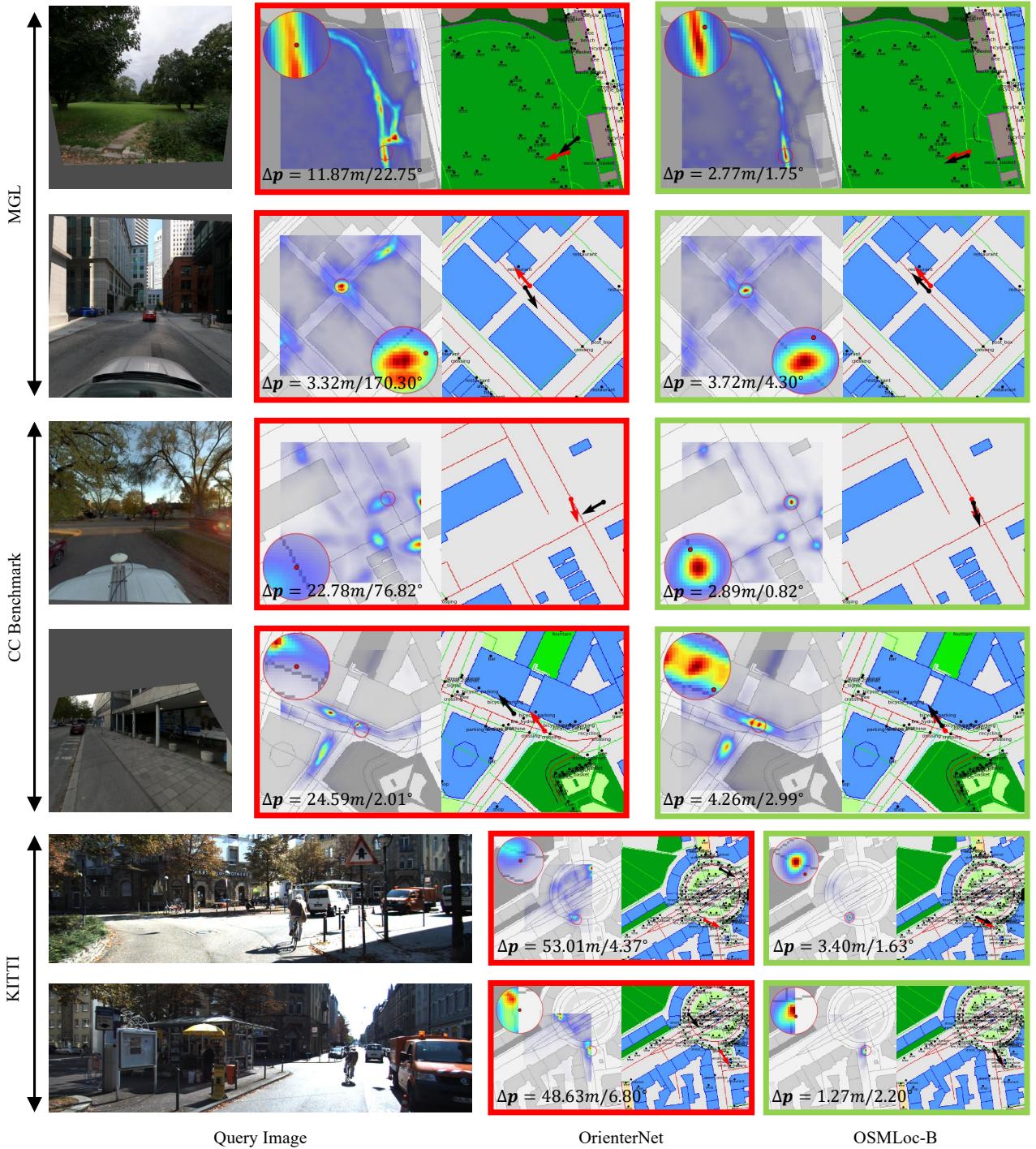


Figure 9: Qualitative results on the MGL dataset, CC validation benchmark and KITTI dataset. The red rectangle \square represents the wrong localization result and the green rectangle \square represents the correct localization result. Black arrow \uparrow represents the predicted pose and red arrow \uparrow represents the GT pose in the map.

obstructing the view, which makes it difficult for the model to distinguish the exact pose along the roadside. In the third and fourth cases, the model fails to localize itself with a single image, as the objects in the vehicle’s forward direction are extremely similar. Sequential localization with the motion pattern can greatly improve the accuracy, as discussed in Sec. 5.4.

5.4. Application in sequential localization

From observed failures, we learn that while our method enhances single-image I2O localization performance, the task remains inherently ill-posed and uncertain due to limited FoV information and scale ambiguity, which may lead to occasional failures. For example, nearby crossings often

Table 5

Ablation studies on the foundation model of image encoder.

Image Encoder	PR @Xm↑			OR @X°↑		
	@1m	@3m	@5m	@1°	@3°	@5°
ResNet-101	12.90	43.27	56.05	20.48	52.28	66.42
DINOv2-S	14.04	44.00	57.31	21.84	53.38	68.41
DINOv2-B	15.30	46.05	59.40	21.79	56.57	71.29

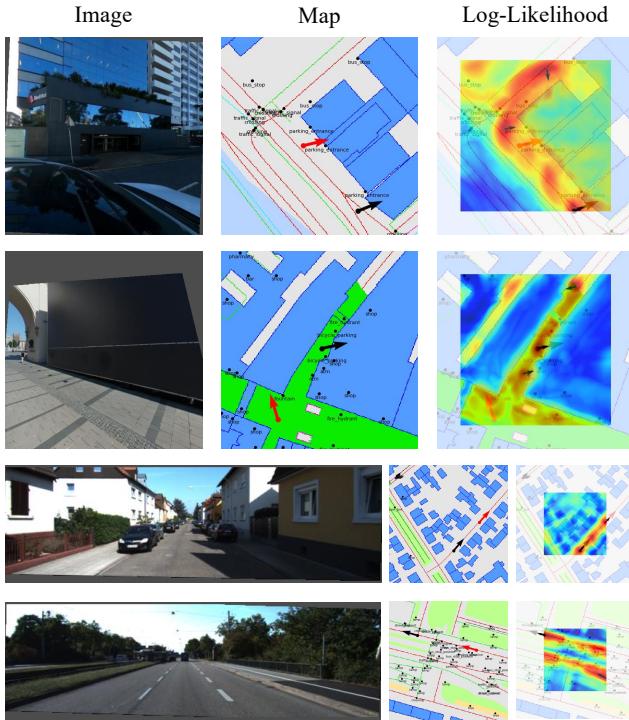


Figure 10: Failure cases. The first and second ones are from the MGL dataset and the CC validation benchmark, while the third and last ones are from the KITTI dataset. Black arrow ↑ represents the predicted pose and red arrow ↑ represents the GT pose in the map.

appear similar and are difficult to distinguish, even for human users with a single glance. Leveraging sequential images could effectively mitigate these issues, and our method serves as a robust observation model for existing sequence-based approaches.

Following previous approaches (Chen et al., 2021; Zhou et al., 2021), we utilize the Monte Carlo Localization (MCL) framework with the particle filter to solve the sequential localization problem. For the first frame, we initialize N particles onto the top- N pose candidates and each particle represents a pose hypothesis $\mathbf{p}_1^i = (x_1^i, y_1^i, \theta_1^i)$. When the camera moves, the pose of each particle is updated based on the motion model with the control input \mathbf{u} . The expected observation from the predicted pose of each particle is then compared with the actual observation acquired by the camera to update the weight of each sample. Particles are resampled based on the weight distribution and reduced for efficiency when the effective sample number is less than a predefined threshold. After several iterations, the particles gradually converge around the GT pose.

At timestamp t , MCL realizes a recursive Bayesian filter estimating the probability density $p(\mathbf{p}_t^i | \mathcal{S}_{1:t}, \mathbf{u}_{1:t})$ over the pose \mathbf{p}_t given all observation $\mathcal{S}_{1:t}$ and motion inputs $\mathbf{u}_{1:t}$ up to time t . Here \mathcal{S} is the matching score volume, and t is reused as the timestamp for simplicity. The posterior is updated as:

$$p(\mathbf{p}_t | \mathcal{S}_{1:t}, \mathbf{u}_{1:t}) = \eta p(\mathcal{S}_t | \mathbf{p}_t, \mathcal{M}) \quad (13)$$

$$\int p(\mathbf{p}_t | \mathbf{u}_t, \mathbf{p}_{t-1}) p(\mathbf{p}_{t-1} | \mathcal{S}_{1:t-1}, \mathbf{u}_{1:t-1}) d\mathbf{p}_{t-1}, \quad (14)$$

where η is the normalized constant, $p(\mathbf{p}_t | \mathbf{u}_t, \mathbf{p}_{t-1})$ is the motion model, and the $p(\mathcal{S}_t | \mathbf{p}_t, \mathcal{M})$ is the observation model. We use the standard motion model in (Thrun, 2002) and focus on the observation model. For each particle i , we query the matching score $\mathcal{S}_t(x_t^i, y_t^i, \theta_t^i)$ from the score volume and approximate the likelihood $p(\mathcal{S}_t | \mathbf{p}_t^i, \mathcal{M})$ as a Gaussian observation model:

$$p(\mathcal{S}_t | \mathbf{p}_t^i, \mathcal{M}) \propto w_t^i = \exp\left(\frac{\log \mathcal{S}_t(x_t^i, y_t^i, \theta_t^i)}{2\zeta^2}\right), \quad (15)$$

where ζ controls the sensitivity of the observation. With the weight w_t^i of each particle, the posterior pose $\hat{\mathbf{p}}_t$ is computed as the weighted average vector of the candidate poses:

$$\hat{\mathbf{p}}_t = \frac{\sum_{i=1}^N w_t^i \mathbf{p}_t^i}{\sum_{i=1}^N w_t^i}. \quad (16)$$

During implementation, the number of particles is set to $N = 1000$ at initialization and reduced to 200 if converged. The sensitivity parameter ζ is set to 2.0. The minimal and maximal sequence lengths are set to 3 and 10 respectively. To simulate the real motion pattern, random Gaussian noise ϵ_p is added to the GT pose. For the MGL dataset and CC validation benchmark, we sub-sample the sequences and ensure that frames are spaced by at least 4m, and $\epsilon_x, \epsilon_y \sim \mathcal{N}(0, 0.5m)$, $\epsilon_\theta \sim \mathcal{N}(0, 0.1\pi)$. For the KITTI dataset, we use the raw sequences and $\epsilon_x, \epsilon_y \sim \mathcal{N}(0, 0.05m)$, $\epsilon_\theta \sim \mathcal{N}(0, 0.01\pi)$.

The quantitative results of sequential localization on the MGL dataset, CC validation benchmark, and KITTI dataset are presented in Table. 7 and Table. 8. With the identical motion model and experimental settings, our method achieves higher position recall (PR) and orientation recall (OR) values than the baseline method. Compared with single-frame localization results, sequential localization significantly improves the accuracy and reduces ambiguity, as shown in Fig. 11.

6. Conclusion

In this paper, we present the OSMLoc, a brain-inspired single image-to-OpenStreetMap (I2O) visual localization framework. The proposed method integrates the recent foundation model Depth Anything for geometric guidance, and fully exploits semantic information from the OSM data to enhance the perception capability of the image model. A

Table 6

Ablation studies on depth guidance and semantic guidance. OSMLoc-S is selected as the baseline. The \mathcal{L}_{pose} means the pose classification loss. \mathcal{L}_{depth} means the depth distribution loss. The \mathcal{L}_{sem} means the semantic alignment loss.

	\mathcal{L}_{pose}	\mathcal{L}_{depth}	\mathcal{L}_{sem}	PR @Xm↑			OR @X°↑		
				@1m	@3m	@5m	@1°	@3°	@5°
[A]	✓			11.52	42.54	55.53	20.80	51.44	65.79
[B]	✓	✓		12.33	43.22	55.15	21.22	51.74	66.01
[C]	✓		✓	13.20	43.43	56.16	20.43	51.08	66.16
[D]	✓	✓	✓	14.04	44.00	57.31	21.84	53.38	68.41

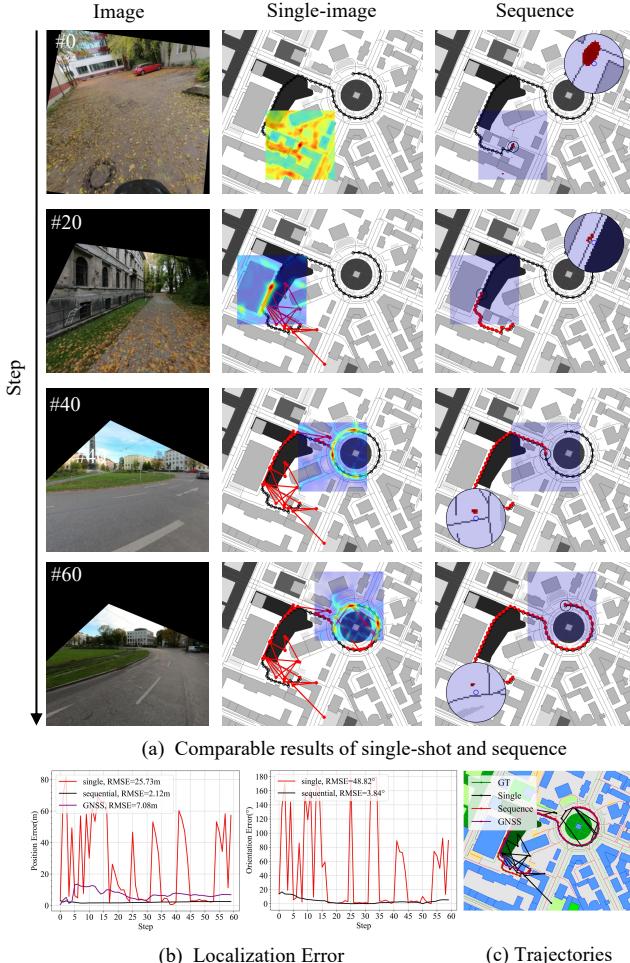


Figure 11: Qualitative results of sequential localization in Munich. (a) comparable localization results of single-image and sequential images. (b) Incorporating multi-frame observation and the motion pattern reduces the ambiguity and consistently improves the accuracy and robustness.

novel cross-area and cross-condition (CC) validation benchmark is proposed to evaluate the accuracy, robustness and generalizability of I2O visual localization methods. Experiments on the MGL dataset and CC validation benchmark have demonstrated that our OSMLoc achieves higher localization accuracy, while extensive experiments on the KITTI dataset highlight its application value in the autonomous driving area. Sequential localization experiments underscore

Table 7

Sequential localization results on the MGL dataset and CC validation benchmark. We use **bold** font to indicate the best results. "Detroit", "Munich", "Taipei" and "Brisbane" mean the city subsets of the CC validation benchmark. "single" means the single-image localization and "seq" means the sequential localization. "Ori" is short for OrienterNet.

Data	Setup	Method	PR @Xm↑			OR @X°↑		
			@1m	@3m	@5m	@1°	@3°	@5°
MGL	single	Ori	10.69	42.12	54.01	19.22	48.56	63.59
		Ours	15.30	46.05	59.40	21.79	56.57	71.29
	seq	Ori	21.47	59.56	72.90	22.32	56.70	71.47
		Ours	25.13	61.74	77.42	25.77	61.64	75.88
Detroit	single	Ori	7.64	33.11	46.24	13.66	32.10	44.57
		Ours	13.37	51.79	63.01	19.75	48.57	62.41
	seq	Ori	15.41	53.60	66.13	16.79	42.39	57.01
		Ours	23.74	71.16	80.04	23.56	59.71	74.22
Munich	single	Ori	2.68	17.90	35.11	12.48	32.75	47.40
		Ours	4.36	22.03	40.58	15.35	41.79	59.65
	seq	Ori	4.14	31.92	54.41	15.99	43.45	58.78
		Ours	7.99	40.57	63.10	21.19	53.07	71.79
Taipei	single	Ori	1.23	8.17	16.13	5.88	16.40	24.79
		Ours	1.44	9.72	19.18	7.32	21.31	32.10
	seq	Ori	2.62	15.36	26.18	5.30	16.54	25.80
		Ours	2.74	16.10	29.60	6.37	21.31	32.82
Brisbane	single	Ori	0.21	2.60	7.47	4.66	9.73	17.60
		Ours	0.27	2.88	9.10	5.00	15.34	24.86
	seq	Ori	1.03	7.16	16.39	4.41	12.74	20.59
		Ours	1.38	9.16	17.70	6.54	16.74	26.93

the potential practical value. We hope that our OSMLoc will benefit the relevant communities in the field.

7. Acknowledgment

This study is supported by the National Natural Science Foundation Project (No. 42201477, No. 42130105).

References

- Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J., 2016. Netvlad: Cnn architecture for weakly supervised place recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5297–5307.
- Barsan, I.A., Wang, S., Pokrovsky, A., Urtasun, R., 2018. Learning to localize using a lidar intensity map, in: Conference on Robot Learning (CoRL).
- Chen, K., Yu, H., Yang, W., Yu, L., Scherer, S., Xia, G.S., 2022. I2d-loc: Camera localization via image to lidar depth flow. ISPRS Journal of Photogrammetry and Remote Sensing 194, 209–221.
- Chen, X., Vizzo, I., Läbe, T., Behley, J., Stachniss, C., 2021. Range image-based lidar localization for autonomous vehicles, in: 2021 IEEE

Table 8

Sequential localization results on the KITTI dataset. We use **bold** font to indicate the best results. "single" means the single-image localization and "seq" means the sequential localization.

Method	Setup	LatR @Xm↑			LonR @Xm↑			OR @X°↑		
		@1m	@3m	@5m	@1m	@3m	@5m	@1°	@3°	@5°
OrienterNet	single	62.03	90.89	95.73	19.99	51.60	65.78	29.93	72.81	87.79
Ours		66.71	94.26	97.04	27.14	62.09	73.80	37.43	79.69	92.61
OrienterNet	seq	66.86	93.39	97.16	22.84	59.79	75.50	45.75	87.55	95.43
Ours		74.92	96.26	98.47	30.18	71.57	83.78	49.58	91.19	97.62

- International Conference on Robotics and Automation (ICRA), IEEE. pp. 5802–5808.
- Cheng, L., Yuan, Y., Xia, N., Chen, S., Chen, Y., Yang, K., Ma, L., Li, M., 2018. Crowd-sourced pictures geo-localization method based on street view images and 3d reconstruction. ISPRS journal of photogrammetry and remote sensing 141, 72–85.
- Cheng, W., Lin, W., Zhang, X., Goesele, M., Sun, M.T., 2016. A data-driven point cloud simplification framework for city-scale image-based localization. IEEE Transactions on Image Processing 26, 262–275.
- Cho, Y., Kim, G., Lee, S., Ryu, J.H., 2022. Openstreetmap-based lidar global localization in urban environment without a prior lidar map. IEEE Robotics and Automation Letters 7, 4999–5006.
- Dosovitskiy, A., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 .
- Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., Sattler, T., 2019. D2-net: A trainable cnn for joint description and detection of local features, in: Proceedings of the ieee/cvf conference on computer vision and pattern recognition, pp. 8092–8101.
- Epstein, R.A., Vass, L.K., 2014. Neural systems for landmark-based wayfinding in humans. Philosophical Transactions of the Royal Society B: Biological Sciences 369, 20120533.
- Fan, H., Zipf, A., Fu, Q., Neis, P., 2014. Quality assessment for building footprints data on openstreetmap. International Journal of Geographical Information Science 28, 700–719.
- Floros, G., Van Der Zander, B., Leibe, B., 2013. Openstreetslam: Global vehicle localization using openstreetmaps, in: 2013 IEEE international conference on robotics and automation, IEEE. pp. 1054–1059.
- Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? the kitti vision benchmark suite, in: 2012 IEEE conference on computer vision and pattern recognition, IEEE. pp. 3354–3361.
- Google, 2024. Google map. URL: <https://www.google.com/maps>.
- Hausler, S., Garg, S., Xu, M., Milford, M., Fischer, T., 2021. Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 14141–14152.
- Hermer, L., Spelke, E.S., 1994. A geometric process for spatial reorientation in young children. Nature 370, 57–59.
- Huitl, R., Schroth, G., Hilsenbeck, S., Schweiger, F., Steinbach, E., 2012. Tumindoor: An extensive image and point cloud dataset for visual indoor localization and mapping, in: 2012 Ninth IEEE International Conference on Image Processing, IEEE. pp. 1773–1776.
- Kang, S., Liao, Y., Li, J., Liang, F., Li, Y., Zou, X., Li, F., Chen, X., Dong, Z., Yang, B., 2024. Cofi2p: Coarse-to-fine correspondences-based image to point cloud registration. IEEE Robotics and Automation Letters.
- Keetha, N., Mishra, A., Karhade, J., Jatavallabhula, K.M., Scherer, S., Krishna, M., Garg, S., 2023. Anyloc: Towards universal visual place recognition. IEEE Robotics and Automation Letters.
- Kendall, A., Grimes, M., Cipolla, R., 2015. Posenet: A convolutional network for real-time 6-dof camera relocalization, in: Proceedings of the IEEE international conference on computer vision, pp. 2938–2946.
- Lee, S., Ryu, J.H., 2024. Autonomous vehicle localization without prior high-definition map. IEEE Transactions on Robotics .
- Lentsch, T., Xia, Z., Caesar, H., Kooij, J.F., 2023. Slicematch: Geometry-guided aggregation for cross-view pose estimation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21632–21642.
- Sattler, T., Leibe, B., Kobbelt, L., 2011. Fast image-based localization using direct 2d-to-3d matching, in: 2011 International Conference on Computer Vision, IEEE. pp. 667–674.
- the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 17225–17234.
- Li, G., Qian, M., Xia, G.S., 2024. Unleashing unlabeled data: A paradigm for cross-view geo-localization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16719–16729.
- Li, J., Lee, G.H., 2021. Deepi2p: Image-to-point cloud registration via deep classification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15960–15969.
- Li, Q., Zhu, J., Liu, J., Cao, R., Fu, H., Garibaldi, J.M., Li, Q., Liu, B., Qiu, G., 2020. 3d map-guided single indoor image localization refinement. ISPRS Journal of Photogrammetry and Remote Sensing 161, 13–26.
- Li, Z., Lee, C.D.W., Tung, B.X.L., Huang, Z., Rus, D., Ang, M.H., 2023. Hot-netvlad: Learning discriminatory key points for visual place recognition. IEEE Robotics and Automation Letters 8, 974–980.
- Lin, T.Y., Belongie, S., Hays, J., 2013. Cross-view image geolocalization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 891–898.
- Liu, K., Li, Q., Qiu, G., 2020. Posegan: A pose-to-image translation framework for camera localization. ISPRS Journal of Photogrammetry and Remote Sensing 166, 308–315.
- Liu, Z., Tang, H., Amini, A., Yang, X., Mao, H., Rus, D.L., Han, S., 2023. Bevfusion: Multi-task multi-sensor fusion with unified bird’s-eye view representation, in: 2023 IEEE international conference on robotics and automation (ICRA), IEEE. pp. 2774–2781.
- Mapillary, 2024. Mapillary website. URL: <https://www.mapillary.com/>.
- Oquab, M., Darabet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al., 2023. Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 .
- OSM, 2024. Osm website. URL: <https://www.openstreetmap.org/>.
- OSM_Foundation, 2024. Osm static. URL: https://planet.openstreetmap.org/statistics/data_stats.html.
- Philion, J., Fidler, S., 2020. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16, Springer. pp. 194–210.
- Samano, N., Zhou, M., Calway, A., 2020. You are here: Geolocation by embedding maps and images, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16, Springer. pp. 502–518.
- Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A., 2020. Super glue: Learning feature matching with graph neural networks, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4938–4947.
- Sarlin, P.E., DeTone, D., Yang, T.Y., Avetisyan, A., Straub, J., Malisiewicz, T., Bulo, S.R., Newcombe, R., Kontschieder, P., Balntas, V., 2023. Orienternet: Visual localization in 2d public maps with neural matching, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21632–21642.
- Sarlin, P.E., Trulls, E., Pollefeys, M., Hosang, J., Lynen, S., 2024. Snap: Self-supervised neural maps for visual positioning and semantic understanding. Advances in Neural Information Processing Systems 36.

- Shi, C., Li, J., Gong, J., Yang, B., Zhang, G., 2022. An improved lightweight deep neural network with knowledge distillation for local feature extraction and visual localization using images and lidar point clouds. *ISPRS journal of photogrammetry and remote sensing* 184, 177–188.
- Shi, Y., Li, H., 2022. Beyond cross-view image retrieval: Highly accurate vehicle localization using satellite image, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 17010–17020.
- Tang, S., Li, Y., Wan, J., Li, Y., Zhou, B., Guo, R., Wang, W., Feng, Y., 2024. Transcnnloc: End-to-end pixel-level learning for 2d-to-3d pose estimation in dynamic indoor scenes. *ISPRS Journal of Photogrammetry and Remote Sensing* 207, 218–230.
- Thrun, S., 2002. Probabilistic robotics. *Communications of the ACM* 45, 52–57.
- Tian, Y., Chen, C., Shah, M., 2017. Cross-view image matching for geo-localization in urban environments, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3608–3616.
- Vargas-Munoz, J.E., Srivastava, S., Tuia, D., Falcao, A.X., 2020. Open-streetmap: Challenges and opportunities in machine learning and remote sensing. *IEEE Geoscience and Remote Sensing Magazine* 9, 184–199.
- Vass, L.K., Epstein, R.A., 2013. Abstract representations of location and facing direction in the human brain. *Journal of Neuroscience* 33, 6133–6142.
- Wang, Y., Jiao, W., Fan, H., Zhou, G., 2024. A framework for fully automated reconstruction of semantic building model at urban-scale using textured lod2 data. *ISPRS Journal of Photogrammetry and Remote Sensing* 216, 90–108.
- Warburg, F., Hauberg, S., Lopez-Antequera, M., Gargallo, P., Kuang, Y., Civera, J., 2020. Mapillary street-level sequences: A dataset for lifelong place recognition, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 2626–2635.
- Wu, H., Zhang, Z., Lin, S., Mu, X., Zhao, Q., Yang, M., Qin, T., 2024. Maplocnet: Coarse-to-fine feature registration for visual re-localization in navigation maps. *arXiv preprint arXiv:2407.08561*.
- Xia, Z., Booij, O., Manfredi, M., Kooij, J.F., 2022. Visual cross-view metric localization with dense uncertainty estimates, in: European Conference on Computer Vision, Springer. pp. 90–106.
- Yan, F., Vysotska, O., Stachniss, C., 2019. Global localization on open-streetmap using 4-bit semantic descriptors, in: 2019 European conference on mobile robots (ECMR), IEEE. pp. 1–7.
- Yang, L., Kang, B., Huang, Z., Xu, X., Feng, J., Zhao, H., 2024. Depth anything: Unleashing the power of large-scale unlabeled data, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10371–10381.
- Ye, Q., Luo, J., Lin, Y., 2024. A coarse-to-fine visual geo-localization method for gnss-denied uav with oblique-view imagery. *ISPRS Journal of Photogrammetry and Remote Sensing* 212, 306–322.
- Yu, S., Wang, C., Yu, Z., Li, X., Cheng, M., Zang, Y., 2021. Deep regression for lidar-based localization in dense urban areas. *ISPRS Journal of Photogrammetry and Remote Sensing* 172, 240–252.
- Zhou, M., Chen, X., Samano, N., Stachniss, C., Calway, A., 2021. Efficient localisation using images and openstreetmaps, in: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE. pp. 5507–5513.
- Zhu, S., Yang, L., Chen, C., Shah, M., Shen, X., Wang, H., 2023. R2former: Unified retrieval and reranking transformer for place recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 19370–19380.
- Zhu, S., Yang, T., Chen, C., 2021. Vigor: Cross-view image geo-localization beyond one-to-one retrieval, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3640–3649.
- Zou, X., Li, J., Wang, Y., Liang, F., Wu, W., Wang, H., Yang, B., Dong, Z., 2023. Patchaugnet: Patch feature augmentation-based heterogeneous point cloud place recognition in large-scale street scenes. *ISPRS Journal of Photogrammetry and Remote Sensing* 206, 273–292.