Beyond Cross-view Image Retrieval:
Highly Accurate Vehicle Localization Using Satellite Image

Yujiao Shi and Hongdong Li
The Australian National University, Canberra, Australia
`firstname.lastname@anu.edu.au`

Abstract

This paper addresses the problem of vehicle-mounted camera localization by matching a ground-level image with an overhead-view satellite map. Existing methods often treat this problem as cross-view image retrieval, and use learned deep features to match the ground-level query image to a partition (e.g., a small patch) of the satellite map. By these methods, the localization accuracy is limited by the partitioning density of the satellite map (often in the order of tens meters). Departing from the conventional wisdom of image retrieval, this paper presents a novel solution that can achieve highly-accurate localization. The key idea is to formulate the task as pose estimation and solve it by neural-net based optimization. Specifically, we design a two-branch CNN to extract robust features from the ground and satellite images, respectively. To bridge the vast cross-view domain gap, we resort to a Geometry Projection module that projects features from the satellite map to the ground-view, based on a relative camera pose. Aiming to minimize the differences between the projected features and the observed features, we employ a differentiable Levenberg-Marquardt (LM) module to search for the optimal camera pose iteratively. The entire pipeline is differentiable and runs end-to-end. Extensive experiments on standard autonomous vehicle localization datasets have confirmed the superiority of the proposed method. Notably, e.g., starting from a coarse estimate of camera location within a wide region of $40m \times 40m$, with an 80% likelihood our method quickly reduces the lateral location error to be within $5m$ on a new KITTI cross-view dataset.

(a) Cross-view image retrieval-based localization

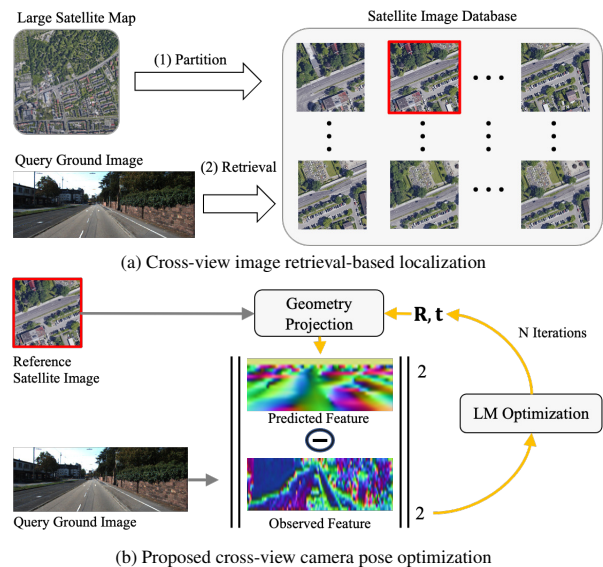(b) Proposed cross-view camera pose optimization

Figure 1. Comparison of two schemes for cross-view camera localization. (a) Conventional image retrieval-based scheme first splits a large satellite map into small image patches and constructs a reference database. Given a query image, they find the most similar image from a database. GPS-tag of the retrieved image is regarded as the query camera's location. By this method, the localization accuracy is limited by the sample density of database images. (b) The paper proposes a novel scheme for cross-view localization, which formulates the task as camera pose optimization. We tentatively project the satellite deep features to a ground viewpoint from an initial camera pose estimate by a Geometry Projection module. Then, a differentiable Levenberg-Marquardt optimization procedure is applied to refine the camera pose estimation by minimizing the differences between the predicted and observed features.

1. Introduction

Image-based camera localization [3, 8, 14, 20, 29, 36, 40, 43, 58] has attracted increasing attention from the community due to its practical applications in various fields, including autonomous driving, virtual and augmented reality. Recently, this technique has been extended to the cross-view

setting, *i.e.*, localization by matching a ground-level image to an overhead-view satellite map to determine the ground camera's pose.

Existing learning-based image localization methods often treat this task as an instance of image retrieval, and solve it by *metric learning* [5,17,27,37,46,47,49,51,55,57,60,63, 64,67,69]. They match a ground image to many candidate

satellite patches in a large satellite map covering the geographical region of interest and then retrieve the most similar one. The query camera's location is then assigned as the GPS tag of the retrieved satellite patch. Although promising results have been achieved, their estimation accuracy is limited by the sampling density of those satellite images. A recent work [70] further discussed location refinement by using a deep network to regress the relative displacement. However, the significant domain differences between the satellite and ground-view images makes it very difficult to obtain accurate regression in the cross-view setting.

Departing from the traditional idea of image retrieval, we propose to solve accurate cross-view localization task by direct pose optimization. Specifically, we first employ two CNNs to extract deep features from the two-view images. The learned features are expected to be robust to view changes and discriminative for feature correspondences. Then, we devise a Geometry Projection module, which approximately projects satellite features to a ground viewpoint, based on the current camera pose estimate, to bridge the domain gap between the views. Finally, a differentiable Levenberg-Marquardt (LM) algorithm is embedded in the pipeline to refine the pose. The LM optimization aims to find the optimal camera pose such that the predicted deep features originated from the satellite image match well to the corresponding deep features extracted from the ground-view image. Please refer to Fig. 1 for an overview.

We evaluate our method on two standard benchmarks for autonomous driving, *i.e.* KITTI [15] and Ford multi-AV datasets [2]. Both the datasets contain ground-level images by a vehicle-mounted camera with GT poses but without satellite maps. We supplement them with corresponding high-definition satellite maps, downloaded from Google Map [1], for the evaluation of the proposed method.

2. Prior Arts

Image based localization. Image-based localization is often formulated as an image retrieval problem and tackled by metric learning techniques. It is solved by ground-to-ground (G2G) image matching [3, 8, 9, 13, 14, 20, 29, 34, 36, 40, 43, 58]. Since the G2G image matching cannot localize query images whose reference counterparts are not available, many recent works resort to the widespread satellite images to construct the database [5, 7, 17, 27, 37, 45–49, 51, 55, 57, 60, 63, 67, 69].

These works approximate the query camera pose as the pose of the top-1 retrieved reference image. They remain effective at scale, but the pose estimates in this manner are very coarse. In this work, we introduce a novel approach to increase localization accuracy.

3D structure based localization. Works on 3D structure based localization usually employ a 3D scene model as reference for query camera localization [4, 6, 10, 21, 23,

24, 28, 30, 39, 41, 42, 52, 53, 56, 66, 68]. Among these algorithms, works [38, 61] also use the LM optimization for camera pose estimation. However, they are designed for ground-to-ground localization only and require knowing the 3D coordinates of image key points. This paper only uses a high-definition satellite image as a reference and solves the ground-to-satellite localization when 3D scene models are unavailable.

SLAM/VO. Simultaneous Localization and Mapping (SLAM) and Visual Odometry (VO) techniques have been traditionally used for vehicle localization [11, 16, 18, 19, 30, 31, 33, 35, 44, 50, 59]. They first estimate relative camera poses between consecutive image frames, and then integrate them for global pose computation. As such, they suffer from error accumulation, leading to an estimation drift. Our proposed method only relies on a single frame. Hence, it can complement the SLAM/VO method as a novel way of (satellite image-based) loop-closure.

3. Method Overview

Given a coarse initial estimate of a ground camera's pose, we aim to optimize this pose in high accuracy by matching it to a companion satellite map. Instead of formulating this task as image retrieval, we propose a pose optimization framework described below.

Our framework consists of three components: (i) a two-branch deep network for feature learning, (ii) a Geometry Projection module, and (iii) a differentiable Levenberg-Marquardt (LM) optimizer, as shown in Fig. 2.

3.1. Deep robust feature learning

Deep neural networks are shown to be powerful in learning robust features that are resilient to viewpoint changes and suitable for the task of visual localization [5, 27, 37, 46, 47, 51, 55, 60, 63, 64]. We design a two-branch neural network to extract deep features from the ground and satellite images separately. The U-net structure is used to learn multi-scale feature representations.

3.2. Cross-view feature alignment

To bridge the evident cross-view domain gap, we devise a Geometry Projection module which aligns the two view features in the ground-view domain based on cross-view geometry. Our geometry projection module projects satellite-view features to a ground viewpoint by establishing approximate geometric correspondences and using a relative camera pose between the two views.

3.3. Iterative pose refinement

We aim to find the optimal ground camera pose such that the projected features from the satellite domain are the most similar to the real observed ground-view features. To
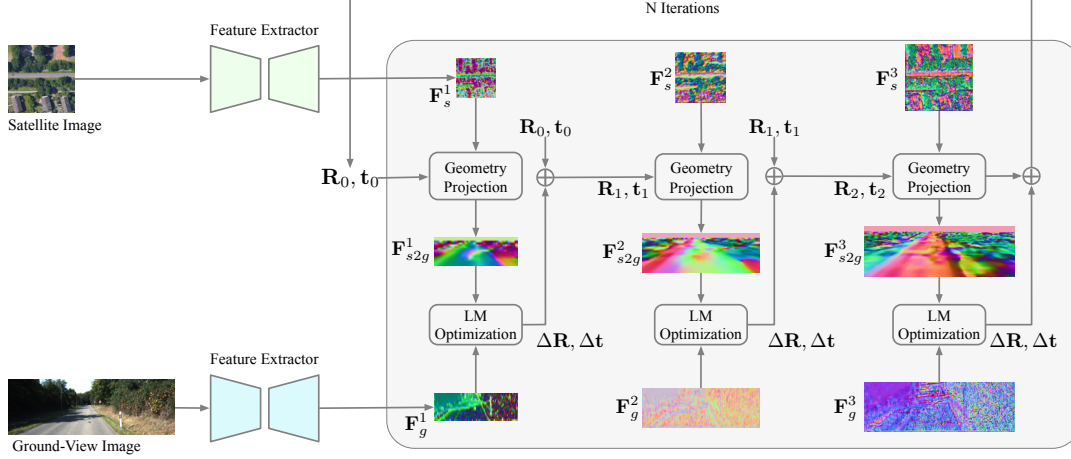
Figure 2. An overview of the proposed highly-accurate camera pose optimization procedure. Our method employs a two-branch deep network to extract multi-scale robust features from ground and satellite images, respectively. Next, the Geometry Projection module maps the satellite features to the ground-view domain, based on an initial camera pose $\mathbf{R}_0, \mathbf{t}_0$. By minimizing the differences between the projected satellite features $\mathbf{F}_{s2g}$ and the observed ground-view features $\mathbf{F}_g$, an LM optimization is employed to find the optimal camera pose. The LM optimization is executed in a coarse to fine manner.

this end, we develop a new differentiable LM optimizer for cross-view feature alignment. The LM optimizer iteratively refines the camera pose to match the cross-view feature maps.

When the initial camera pose is too far from the ground-truth pose, the projected contents from a satellite image will be considerably different from the observed ground image, causing a local minima problem. We apply a coarse-to-fine multi-scale LM update strategy to mitigate this issue. Features at the coarsest level have a larger receptive field in the original image and thus are suitable for coarse-level global search in the solution space. Conversely, features at finer scales have a larger spatial resolution and encode more detailed scene information. Hence, they are more informative for precise pose refinement.

4. Highly-accurate Pose Optimization

In this section, we provide a detailed analysis of the proposed method. As mentioned before, we adopt a two branch CNN to extract deep features from the ground and satellite images, denoted as $\mathbf{F}_g^l \in \mathbb{R}^{H_g^l \times W_g^l \times C^l}$ and $\mathbf{F}_s^l \in \mathbb{R}^{H_s^l \times W_s^l \times C^l}$, respectively, where $l = 1, 2, 3$ indicates the scale level. The ground and satellite branches share the same architecture but do not share weights. In this way, they can adapt to their respective domains. The features at each level are L_2 normalized to increase their robustness for cross-view matching.

4.1. Satellite-to-ground geometry projection

We introduce a Geometry Projection module to establish the cross-view geometric correspondences. Our geometry projection module projects satellite-view features to

a ground viewpoint based on the relative camera pose between the two view images.

We set the world coordinate system to the initial camera pose estimate, with its location corresponds to the reference satellite image center, x axis parallel to the v^s direction of the satellite image, y axis pointing downward, and z axis parallel to the u^s direction. A 3D point (x, y, z) in the world coordinate system is mapped to a satellite pixel coordinate by the orthographic projection,

$$[u_s, v_s]^T = [\frac{z}{\alpha} + u_s^0, \frac{x}{\alpha} + v_s^0]^T, \quad (1)$$

where α as the per-pixel real-word distance of a satellite feature map, and (u_s^0, v_s^0) is the satellite feature map center.

The transformation from the real camera coordinate system to the world coordinate system is formulated as

$$[x, y, z]^T = \mathbf{R}([x_c, y_c, z_c]^T + \mathbf{t}), \quad (2)$$

where $\mathbf{R}$ and $\mathbf{t}$ are the rotation and translation matrices, respectively. The projection from a 3D point to a pin-hole camera image plane is given by

$$w[u^g, v^g, 1]^T = \mathbf{K}[x_c, y_c, z_c]^T, \quad (3)$$

where $\mathbf{K}$ is the camera intrinsic and w is a scale factor.

From Eq. (1)~(3), we can derive the mapping from a ground-view pixel to a satellite pixel as

$$\begin{bmatrix} u_s \\ v_s \\ z \end{bmatrix} = \begin{bmatrix} \frac{1}{\alpha} & 0 & 0 \\ 0 & \frac{1}{\alpha} & 0 \\ 0 & 0 & 1 \end{bmatrix} \left(w\mathbf{R}\mathbf{K}^{-1} \begin{bmatrix} u_g \\ v_g \\ 1 \end{bmatrix} + \mathbf{R}\mathbf{t} \right) + \begin{bmatrix} u_s^0 \\ v_s^0 \\ 0 \end{bmatrix}. \quad (4)$$

When the depth map of the ground-view image is available, i.e., w is given, the satellite to ground projection can

be easily conducted by Eq. (4). However, it is challenging to estimate depths from a single ground image. Considering the overlap between a ground and a satellite image mainly lies on the ground plane, our geometry projection is conducted by using the homography of the ground plane. In other words, we make all the ground-view pixels corresponds to the points on the ground plane by setting y_c in Eq. (3) to the distance between the query camera to the ground plane. Then, w can be computed from Eq. (3).

This projection defined on ground plane homography is only approximately correct. To handle objects higher than the ground place and reduce distortions in the projection, we project deep features rather than RGB pixels to measure the gap. These deep features encode high-level semantic information and therefore are less sensitive to object heights than RGB values. By Eq. (4), we conduct the satellite to ground geometry projection using bilinear interpolation, obtaining $\mathbf{F}_{s2g}^l \in \mathbb{R}^{H_g^l \times W_g^l \times C^l}$ as the projected ground-view features from satellite features at scale l.

4.2. Multi-level LM optimization

The differences between the satellite and the ground observations are given by,

$$\mathbf{e}^l = \mathbf{F}_{s2g}^l - \mathbf{F}_g^l. \qquad (5)$$

The objective is to find the optimal pose $\hat{\mathbf{R}}$ and $\hat{\mathbf{t}}$ of the ground camera by minimizing the following loss function,

$$\hat{\xi} = \arg\min_{\xi} \|\mathbf{e}^l\|_2^2, \qquad (6)$$

where $\|\cdot\|_2^2$ is the L_2 norm, $\xi = \{\mathbf{R}, \mathbf{t}\}$ and $\hat{\xi}$ corresponds to its optimal solution. We solve this non-linear least squares problem by the Levenberg-Marquardt(LM) optimization algorithm [22, 32].

For each level l, we compute a Jacobian matrix and a Hessian matrix,

$$\mathbf{J} = \frac{\partial \mathbf{F}_{s2g}}{\partial \xi} = \frac{\partial \mathbf{F}_{s2g}}{\partial \mathbf{p}_s} \frac{\partial \mathbf{p}_s}{\partial \xi}, \quad \text{and} \quad \mathbf{H} = \mathbf{J}^T \mathbf{J}, \qquad (7)$$

where $\mathbf{p}_s$ is the satellite feature map coordinates.

We choose to use Levenberg's damping formula $i.e.$, $\widetilde{\mathbf{H}} = \mathbf{H} + \lambda \mathbf{I}$, for its convenience in network training. λ is the trade-off parameter which interpolates between the gradient decent ($\lambda = \infty$) and Gaussian-Newton ($\lambda = 0$) algorithm. Alternatively, the Marquardt's damping formula may be used instead [32].

The pose is updated by,

$$\xi_{t+1} = \xi_t + \widetilde{\mathbf{H}}^{-1} \mathbf{J}^T \mathbf{e}, \qquad (8)$$

where t index iterations.

The LM optimization is first applied at the coarsest feature level and gradually propagates to finer levels. This

coarse-to-fine (C2F) scheme is executed iteratively until it converges or reaches a maximum iteration of 5. This multi-scale C2F procedure offers an opportunity to escape from local minima and is more likely to find the global optimum.

We had attempted to embed a confidence map in Eq. (6). The intention was to give higher weight to salient visual features ($e.g.$, corner points) and lower weight to textureless regions. However, in our experiments, we did not observe consistent improvement across different test sets. Hence this idea is not employed in our current method.

4.3. Training objective

The LM optimization is implemented in a differentiable manner in our pipeline (within a feed-forward pass). The network is trained end-to-end. We use the GT camera poses as our network supervision,

$$\mathcal{L} = \sum_t \sum_l (\|\hat{\mathbf{R}}_t^l - \mathbf{R}^*\|_1 + \|\hat{\mathbf{t}}_t^l - \mathbf{t}^*\|_1), \qquad (9)$$

where $\hat{\mathbf{R}}_t^l$ and $\hat{\mathbf{t}}_t^l$ is the predicted pose by our method at the t_{th} iteration and l_{th} level, $\mathbf{R}^*$ and $\mathbf{t}^*$ is the GT camera pose.

During training, when the camera pose provided by the LM optimization deviates from the GT value, the error will be backpropagated to the feature extraction network and update its parameters. In this way, our network is trained to learn useful cross-view features for pose optimization.

5. Satellite-augmented KITTI and Ford Multi-AV dataset

We evaluate the feasibility of the proposed method in two standard autonomous driving datasets,$i.e.$, KITTI [15] and Ford multi-AV dataset [2]. Cameras used in both dataset are intrinsically calibrated. Based on the GPS provided by the datasets, we collect satellite images from Google Map [1]. The satellite images are downloaded with zoom 18. The per-pixel resolution for satellite images in KITTI is 0.20 m, and for Ford multi-AV, it is 0.22 m.

KITTI. The KITTI dataset contains stereo images captured by a moving vehicle from different trajectories at different times. There is barely any revisited trajectory. We split the entire dataset (raw data) into three subsets, one for training and two for testing, denoted as Training, Test1, and Test2, respectively. The Training and Test1 sets are from the same region, while Test2 is in a different area. Test2 is used to evaluate the generalization ability of an algorithm. We use the left image in a stereo pair as our query image.

Ford multi-AV dataset. The Ford multi-AV dataset consists of data captured by three vehicles, V1, V2, and V3. Each vehicle is equipped with 7 cameras. Among the three vehicles, only V2 captured images from six trajectories (Log1~Log6) at two different dates/drives, $i.e.$, 2017-08-04 and 2017-10-26. Hence, we use the front left camera of
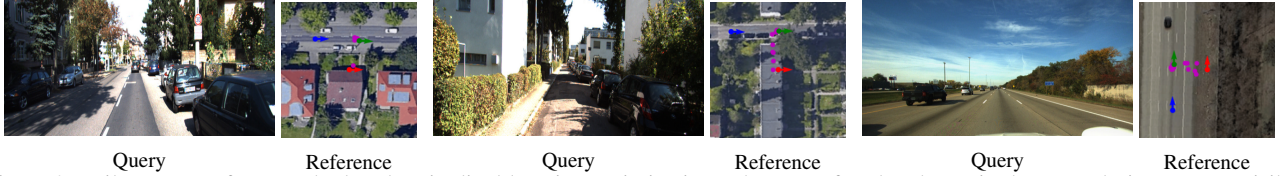
Figure 3. Failure cases of our method on longitudinal location optimization. The scene facades shown in the ground-view are not visible in the overhead view. We can only determine the lateral translation of a ground vehicle. Driving along a highway suffers even considerable ambiguity on the longitudinal location optimization. Here, red arrows represent initial poses, green arrows denote final predicted poses, and blue arrows indicate GT poses. The magenta points pinpoint intermediate locations during optimization.

Table 1. Performance comparison between our method and state-of-the-art methods on KITTI dataset.

| | Test1 | | | | | | | | | Test2 | | | | | | | | |
| | Lateral | | | Longitudinal | | | Azimuth | | | Lateral | | | Longitudinal | | | Azimuth | | |
	$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$	$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$
CVM-NET [17]	5.83	17.41	28.78	3.47	11.18	18.42	-	-	-	6.96	21.55	35.24	3.58	10.45	17.53	-	-	-
CVFT [49]	7.71	22.37	36.28	3.82	11.48	18.63	-	-	-	7.20	22.05	36.21	3.63	11.11	18.46	-	-	-
SAFA [46]	9.49	29.31	46.44	4.35	12.46	21.10	-	-	-	9.15	27.83	44.27	4.22	11.93	19.65	-	-	-
Polar-SAFA [46]	9.57	30.08	45.83	4.56	13.01	21.12	-	-	-	10.02	29.09	46.19	3.82	11.87	19.84	-	-	-
DSM [47]	10.12	30.67	48.24	4.08	12.01	20.14	3.58	13.81	24.44	10.77	31.37	48.24	3.87	11.73	19.50	3.53	14.09	23.95
VIGOR [70]	18.61	49.06	69.79	4.29	13.01	21.47	-	-	-	17.38	48.20	70.79	4.07	12.52	20.14	-	-	-
Ours	**35.54**	**70.77**	**80.36**	**5.22**	**15.88**	**26.13**	**19.64**	**51.76**	**71.72**	**27.82**	**59.79**	**72.89**	**5.75**	**16.36**	**26.48**	**18.42**	**49.72**	**71.00**

Table 2. Performance comparison between our method and state-of-the-art image retrieval approaches on the Ford multi-AV dataset.

| | Log1 | | | | | | | | | Log2 | | | | | | | | |
| | Lateral | | | Longitudinal | | | Azimuth | | | Lateral | | | Longitudinal | | | Azimuth | | |
	$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$	$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$
CVM-NET [17]	9.14	25.67	41.33	4.81	13.19	21.90	-	-	-	9.82	28.60	47.06	4.24	11.83	20.34	-	-	-
CVFT [49]	10.57	31.10	51.19	3.52	11.43	20.38	-	-	-	12.21	35.07	57.61	4.40	12.18	21.41	-	-	-
SAFA [46]	9.33	28.71	47.95	4.33	11.76	20.14	-	-	-	11.22	34.10	53.39	5.02	13.36	22.89	-	-	-
Polar-SAFA [46]	9.05	28.62	47.10	4.43	12.14	21.10	-	-	-	12.02	35.63	56.21	4.29	12.13	20.28	-	-	-
DSM [47]	12.00	35.29	53.67	4.33	12.48	21.43	3.52	13.33	23.67	8.45	24.85	37.64	3.94	12.24	21.41	2.23	7.67	13.42
VIGOR [70]	20.33	52.48	70.43	**6.19**	16.05	25.76	-	-	-	20.87	54.87	75.64	**5.98**	**16.88**	**27.23**	-	-	-
Ours	**46.10**	**70.38**	**72.90**	5.29	**16.38**	**26.90**	**44.14**	**72.67**	**80.19**	**31.20**	**66.46**	**78.27**	4.80	15.27	25.76	**9.74**	**30.83**	**51.62**

V2 as our query images. For each trajectory, we split testing and training sets based on different drives.

Evaluation Metrics. Satellite images can only provide a vehicle's location and orientation (*i.e.*, azimuth angle) reference. Thus this paper estimates a 3-DoF vehicle pose by ground-to-satellite matching. We report vehicle's location errors along the longitudinal direction (*i.e.*, driving direction) and along the lateral direction, separately. This is because, using satellite map to localization, the uncertainty of vehicle location along the driving direction is often more considerable than that along the lateral direction. For instance, when a vehicle is driving on the road with tall buildings on both sides, the vehicle's location along the driving direction is mainly determined by the building facades appearance. However, facades are not visible from the satellite image. Driving on a highway is another scenario where the ambiguity is significant because the scenes along the driving direction are often monotonous and repetitive–uninformative for localization. In contrast, lateral vehicle location can be obtained more reliably using road boundaries. Moreover, multi-lane freeways in rural areas are almost always visible on a satellite map.

When the estimated translation of a camera is within d m to its GT translation along a direction, it is deemed a correct estimation. When the estimated value of a rotation angle is within θ to its GT value, the estimation is deemed correct. We set d to 1, 3, and 5 respectively, and θ to 1°, 3° and 5° respectively. Since this work focuses on autonomous driving, we did not test on other cross-view datasets (*e.g.*, [67], [27] and [70]), also because they do not provide ground-view heading directions relative to the satellite map, making a meaningful comparison harder.

6. Experiments

We first compare our method with those fine-grained image retrieval methods, and then conduct experiments to analyze each component in our framework.

Implementation details. The satellite image resolution used in our experiments is 512×512, corresponding to a coverage of around 100×100 m^2. We assume the city-scale image retrieval has restricted the camera location to

be in a region of 40m × 40m around the satellite image center. Within this region, we conduct a high-accuracy pose search. The choice of this search region guarantees that the satellite image can provide a reference of at least 30 m visual distance for a query camera, *e.g.*, when a query camera is on the boundary of the region and looking outside. The resolution of ground-view images is 256×1024. The feature-level l corresponds to a scale of $\frac{1}{2^{4-l}}$ with respect to the original image resolution. We use Adam optimizer to train our network with a learning rate of 10^{-4}, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. The network is trained by 2 epochs on a RTX3090 GPU with a batch size of 3. Our method is fully implemented in Pytorch. The pose optimization runtime for a query image is around 500 ms. Unless specifically stated, the rotation noise is set to $20°$ throughout the experiments. The source code and datasets can be accessed at https://github.com/shiyujiao/HighlyAccurate.git.

6.1. Comparison with fine-grained image retrieval

Given a query image, its retrieved satellite counterpart from a city-scale database provides a coarse location estimate of this query image. To refine this pose estimate, a reasonable approach might be splitting the satellite image into small patches further and conducting a fine-grained image retrieval. Hence, we first show the performance of the state-of-the-art cross-view localization algorithms by using this fine-grained image retrieval for camera pose refinement.

Settings. During inference stage for image retrieval based method, we sample a grid within the 40m × 40m search region uniformly and crop corresponding satellite patches centered at the grid points to construct the fine-grained retrieval database. Since our method searches 15 possible solutions, *i.e.*, 3 feature levels×5 iterations, the grid size is set to 4×4 in the fine-grained retrieval for a fair comparison. Note that the regular discretized grids only apply to the inference stage. During training stage, the grids are continuously, randomly, and exhaustively sampled.

Competing models. We compare our method with state of the art CVM-NET [17], CVFT [49], SAFA [46], Polar-SAFA [46], DSM [47], and VIGOR [70]. Among these methods, DSM is the only one that can estimate the orientation of a query camera, while others are restricted to location estimation only. Only VIGOR considers the spatial shifts between a query camera location and its matching satellite image center, and they employ two FC layers to regress the spatial shifts. Toker *et al*. [57] needs a matching ground-level image for each database satellite image to train their generator, which is not available in our fine-grained retrieval setting. Hence, we cannot compare to it. The above competing models were retrained (fine-tuned) on our datasets using their original metric learning procedure.

Results. The comparison results on KITTI and Ford
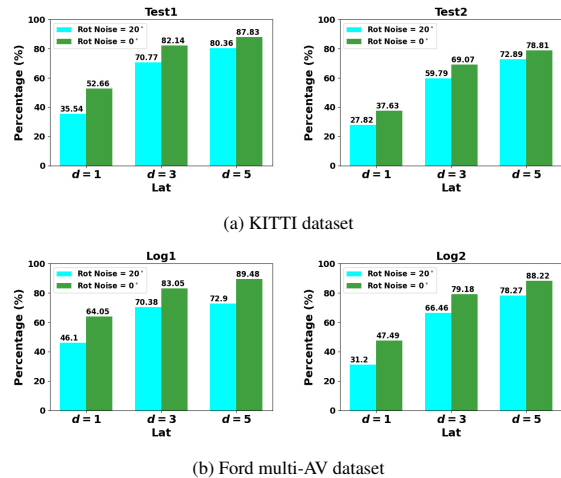

(a) KITTI dataset

(b) Ford multi-AV dataset

Figure 4. Performance comparison of our method when rotation is given or unknown (rotation noise $0°$ Vs. $20°$).

multi-AV dataset are presented in Tab. 1 and Tab. 2, respectively. For space limits, only the results on the first two logs of the Ford multi-AV dataset are presented in the main paper. We provide the performance on the remaining logs of our method in the supplementary material.

From Tab. 1 and Tab. 2, it can be seen that the pure image retrieval-based methods, *i.e.*, CVM-NET [17], CVFT [49], SAFA [46], Polar-SAFA [46], DSM [47], show very poor performance on the high-accuracy distance based localization. This is not only because the database images are discretized but also because the fine-grained partitions of a satellite image are very similar, inducing large uncertainty in cross-view image matching. Since VIGOR explicitly considers the relative displacement between a query camera center and its matching satellite image center, it achieves better performance compared to the pure image retrieval techniques. Moreover, the performance is significantly boosted by using the proposed camera pose optimization mechanism rather than a fine-grained image retrieval.

Visualization. As expected, the performance of all the methods on longitudinal direction is worse than that on the lateral direction. We give some visual examples of such ambiguities along the longitudinal direction in Fig. 3. Despite this ambiguity, our method is still able to refine lateral poses effectively. Fig. 5 gives additional visualizations of the intermediate poses by our method when the scenes are diverse at different regions.

Known orientation. In general, the azimuth rotation of a query camera can be easily obtained from a compass, and the rotation estimation accuracy by SLAM and VO methods is usually accurate. Hence, we test our method when orientation information is given. Fig. 4 reports the test results. It can be seen that the performances of our method are consistently improved.

Figure 5. Visualization of updated poses during optimization. The arrow and point legends are the same as those in Fig. 3.

Table 3. Performance comparison on different ground-and-satellite domain alignment methods on the KITTI dataset.

| | | Test1 | | | | | | | | | Test2 | | | | | | | | |
| | | Lateral | | | Longitudinal | | | Azimuth | | | Lateral | | | Longitudinal | | | Azimuth | | |
		$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$	$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$
G2SP	NN	20.30	53.25	72.12	4.93	15.08	25.31	**21.65**	**54.44**	**71.88**	17.01	46.12	64.41	5.16	15.18	25.31	**20.66**	**51.45**	70.03
	H	27.72	59.98	71.91	**5.75**	**16.80**	**26.13**	18.13	48.77	69.26	25.32	54.63	64.74	4.99	15.61	26.31	17.37	46.57	67.70
S2GP	Polar	18.98	45.93	55.79	5.14	14.95	24.99	13.31	39.25	61.20	11.27	40.51	53.62	4.87	14.73	25.19	13.78	39.68	62.03
	H (**Ours**)	**35.54**	**70.77**	**80.36**	5.22	15.88	**26.13**	19.64	51.76	71.72	**27.82**	**59.79**	**72.89**	**5.75**	**16.36**	**26.48**	18.42	49.72	**71.00**

6.2. Method Analysis

6.2.1 Effectiveness of GP geometry projection

S2GP Vs. G2SP. Compared to satellite images, ground-view images have a larger resolution of scene objects. A small change in the camera pose will be magnified on the appearance changes of ground-view images. In contrast, the corresponding appearance changes in the overhead view are smaller. This sensitiveness of ground-view observations to camera pose changes is a desired property. It contributes to a higher accuracy of estimated poses. Hence, we conduct satellite-to-ground projection (S2GP), instead of ground-to-satellite projection (G2SP), in our geometry projection module. Below, we compare the performance of the two projection methods. Since we use the homography of the ground plane in the projection, we label them by "H" in Tab. 3. As expected, the performance of S2GP is superior to that of G2SP.

Homography Vs. Polar transform. Apart from homography, polar transform was adopted in the literature for bridging the cross-view domain gap [46, 47, 57]. We intend to compare our method with the polar transform in S2GP, labeled as "Polar". From the results in Tab. 3, it can be seen the polar transform performs worse than the homography. This is possiblly because the polar transform only accounts for ground-level panoramas rather than images captured by a ground-level pin-hole camera.

Explicit geometry projection Vs. Implicit network. In contrast to using an explicit geometry transform, we also tested whether a simple neural network can learn an implicit geometric mapping for the same purpose, denoted as "NN".

Here, the NN ablation cannot be conducted in the S2GP direction, because the S2GP is a whole-to-part mapping and will loss much information in the projection. When the initial pose is significantly different with the real one, the synthesized ground-view feature map at the initial pose can be totally different with the real observed one. In this case, it is impossible to make the synthetic and real feature maps aligned by simple image/feature level rotation and translation. Suppose we let NN regenerate a ground feature map for the original satellite image at each pose update. The Jacobian of network parameters will be required at each LM optimization step, which takes a significant amount of GPU memory and is far beyond the capacity of existing 12/24G GPUs. In contrast, our geometry-based S2GP does not involve any network parameters in the LM optimization and thus is feasible. Therefore, our NN ablation is conducted in the G2SP direction. We employ a network that takes a ground image as input and outputs a synthetic satellite feature map. After that, the LM directly rotates and translates the synthetic feature map to register it with its real counterpart without any NN regeneration.

As shown in the first row of Tab. 3, the results are not favorable. Although "NN" achieves slightly better performance than our geometry projection (both S2GP and G2SP) on rotation estimation, its capability in handling translation optimization is rather limited. Our reflection on this is that whenever explicit and principle geometric knowledge about the problem at hand is known and can be used, one should use it instead of black-box neural network implementations.

Table 4. Performance comparison by using different optimizers on the KITTI dataset.

| | Test1 | | | | | | | | | Test2 | | | | | | | | |
| | Lateral | | | Longitudinal | | | Azimuth | | | Lateral | | | Longitudinal | | | Azimuth | | |
	$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$	$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$
SGD	16.86	39.60	51.15	4.72	15.29	25.39	10.05	30.37	49.80	16.06	38.41	50.29	5.00	15.34	25.70	9.98	30.03	50.13
ADAM	7.13	21.15	32.97	4.96	15.13	25.63	10.36	30.32	50.49	7.33	21.36	33.52	5.64	15.38	26.00	10.28	30.81	50.91
Net	27.14	58.28	71.91	4.53	15.19	25.36	**45.56**	**93.19**	**99.76**	20.26	53.94	67.42	5.40	15.82	25.58	**42.03**	**92.32**	**99.81**
LM (**Ours**)	**35.54**	**70.77**	**80.36**	**5.22**	**15.88**	**26.13**	19.64	51.76	71.72	**27.82**	**59.79**	**72.89**	**5.75**	**16.36**	**26.48**	18.42	49.72	71.00

6.2.2 Superiority of LM optimization

LM Vs. SGD and ADAM. Stochastic Gradient Decent (SGD) and ADAM are widely-used optimization methods in neural network training. They have also been demonstrated as effective in many recent Nerf-based methods for scene-specific camera pose estimation [26, 62, 65]. Hence, we compare the LM algorithm employed in this paper with the first-order SGD and Adam on the ground-to-satellite camera pose optimization.

As shown in Tab. 4, it can be seen that the LM optimization performs significantly better than SGD and ADAM. This is because the adaptive second-order LM optimization, as a variant of Gaussian Newton, is essentially guaranteed to find at least one local minimum of a cost function. In contrast, SGD suffers the usual zigzagging behavior and is thus very slow to converge. Although ADAM is often better than SGD on neural network training, we found that it performs the worst among the comparison optimizers in this ground-to-satellite pose optimization.

LM Vs. Network-based optimizer. Using a network to mimic an optimizer has also been investigated in various tasks, for example, optical flow [54], view synthesis [12], and object pose estimation [25]. Hence, we also compare with a network-based optimizer, denoted as "Net" in Tab. 4. The network-based optimizer is composed of a set of convolutional layers and fully connected layers. We also used convolutional GRU and LSTM to construct the network-based optimizer, but we found no significant difference.

Interestingly, we found that the network-based optimizer performs significantly better on rotation optimization while achieving inferior translation optimization performance than LM. This is probably because regular CNNs are not inherently rotation-invariant. A slight rotation change in the input signal will lead to a big difference in the CNN feature maps. Such amplified changes give the CNN-based optimizer more power to search for better rotations. On the other hand, CNNs are translational invariant/equivariant. A small change in translation can be absorbed by higher-level CNN features, adversely affecting the accuracy of translation estimation. These observations have been confirmed in our experiments as seen in Tab. 3. Using a network for the ground-and-satellite domain mapping performs better than the geometry-guided method on rotation optimization, while worse on translation optimization. It deserves further exploration to better combine the advantages of principled theories (*e.g.*, geometry and LM optimization) with data-driven approaches.

7. Conclusion

In this paper, we have proposed a novel method for accurate camera localization using ground-to-satellite cross-view images. This new method represents a departure from the conventional wisdom of image retrieval-based localization. The key challenge lies in properly handling the vast domain gap between the cross-view setting (satellite Vs. ground-view). To this end, we have devised a Geometry Projection module that aligns the two-view features in the ground domain. A principled LM optimization algorithm is employed to optimize the relative camera poses progressively in an end-to-end manner.

Although this work was motivated by the poor accuracy of conventional image retrieval-based localization, we do not intend to replace the image retrieval-based localization technique. Instead, city-scale place retrieval can provide an initial estimate for a query camera. Our method then refines this pose estimate to higher accuracy.

Our ground-to-satellite pose optimization method can also help the conventional SLAM and visual odometry methods for camera tracking as a novel mechanism for "loop closure" in SLAM. In particular, we remark that combining our method with a VO pipeline may resolve the longitudinal ambiguity issue, achieving all-around highly accurate vehicle localization. Furthermore, we expect the overall performance of our method will be further improved when depth information is available, for example, provided by stereo images or Lidar points. This is left as a future work.

8. Acknowledgments

References

[1] https://developers.google.com/maps/documentation/maps-static/overview. 2, 4

[2] Siddharth Agarwal, Ankit Vora, Gaurav Pandey, Wayne Williams, Helen Kourous, and James McBride. Ford multi-av seasonal dataset, 2020. 2, 4

[3] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5297–5307, 2016. 1, 2

[4] Jan Brejcha, Michal Lukáč, Yannick Hold-Geoffroy, Oliver Wang, and Martin Čadík. Landscapear: Large scale outdoor augmented reality by matching photographs with terrain models using learned descriptors. In *European Conference on Computer Vision*, pages 295–312. Springer, 2020. 2

[5] Sudong Cai, Yulan Guo, Salman Khan, Jiwei Hu, and Gongjian Wen. Ground-to-aerial image geo-localization with a hard exemplar reweighting triplet loss. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 1, 2

[6] Song Cao and Noah Snavely. Minimal scene descriptions from structure from motion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 461–468, 2014. 2

[7] Francesco Castaldo, Amir Zamir, Roland Angst, Francesco Palmieri, and Silvio Savarese. Semantic cross-view matching. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 9–17, 2015. 2

[8] Zetao Chen, Adam Jacobson, Niko Sünderhauf, Ben Upcroft, Lingqiao Liu, Chunhua Shen, Ian Reid, and Michael Milford. Deep learning features at scale for visual place recognition. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3223–3230. IEEE, 2017. 1, 2

[9] Mark Cummins and Paul Newman. Fab-map: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research*, 27(6):647–665, 2008. 2

[10] Michael Donoser and Dieter Schmalstieg. Discriminative feature-to-point matching in image-based localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 516–523, 2014. 2

[11] Ryan C DuToit, Joel A Hesch, Esha D Nerurkar, and Stergios I Roumeliotis. Consistent map-based 3d localization on mobile devices. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 6253–6260. IEEE, 2017. 2

[12] John Flynn, Michael Broxton, Paul Debevec, Matthew DuVall, Graham Fyffe, Ryan Overbeck, Noah Snavely, and Richard Tucker. Deepview: View synthesis with learned gradient descent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2367–2376, 2019. 8

[13] Dorian Gálvez-López and Juan D Tardos. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, 28(5):1188–1197, 2012. 2

[14] Yixiao Ge, Haibo Wang, Feng Zhu, Rui Zhao, and Hongsheng Li. Self-supervising fine-grained region similarities for large-scale image localization. In *European Conference on Computer Vision*, pages 369–386. Springer, 2020. 1, 2

[15] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2, 4

[16] Marcel Geppert, Peidong Liu, Zhaopeng Cui, Marc Pollefeys, and Torsten Sattler. Efficient 2d-3d matching for multi-camera visual localization. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 5972–5978. IEEE, 2019. 2

[17] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2, 5, 6

[18] Sixing Hu and Gim Hee Lee. Image-based geo-localization using satellite imagery. *International Journal of Computer Vision*, 128(5):1205–1219, 2020. 2

[19] Eagle S Jones and Stefano Soatto. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *The International Journal of Robotics Research*, 30(4):407–430, 2011. 2

[20] Hyo Jin Kim, Enrique Dunn, and Jan-Michael Frahm. Learned contextual feature reweighting for image geo-localization. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3251–3260. IEEE, 2017. 1, 2

[21] Viktor Larsson, Johan Fredriksson, Carl Toft, and Fredrik Kahl. Outlier rejection for absolute pose estimation with known orientation. In *BMVC*, 2016. 2

[22] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944. 4

[23] Yunpeng Li, Noah Snavely, Dan Huttenlocher, and Pascal Fua. Worldwide pose estimation using 3d point clouds. In *European conference on computer vision*, pages 15–29. Springer, 2012. 2

[24] Yunpeng Li, Noah Snavely, and Daniel P Huttenlocher. Location recognition using prioritized feature matching. In *European conference on computer vision*, pages 791–804. Springer, 2010. 2

[25] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 683–698, 2018. 8

[26] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. *arXiv preprint arXiv:2104.06405*, 2021. 8

[27] Liu Liu and Hongdong Li. Lending orientation to neural networks for cross-view geo-localization. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 5

[28] Liu Liu, Hongdong Li, and Yuchao Dai. Stochastic attraction-repulsion embedding for large scale image localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2570–2579, 2019. 2

[29] Stephanie Lowry, Niko Sünderhauf, Paul Newman, John J Leonard, David Cox, Peter Corke, and Michael J Milford. Visual place recognition: A survey. *IEEE Transactions on Robotics*, 32(1):1–19, 2015. 1, 2

[30] Simon Lynen, Torsten Sattler, Michael Bosse, Joel A Hesch, Marc Pollefeys, and Roland Siegwart. Get out of my lab: Large-scale, real-time visual-inertial localization. In *Robotics: Science and Systems*, volume 1, page 1, 2015. 2

[31] Simon Lynen, Bernhard Zeisl, Dror Aiger, Michael Bosse, Joel Hesch, Marc Pollefeys, Roland Siegwart, and Torsten Sattler. Large-scale, real-time visual–inertial localization revisited. *The International Journal of Robotics Research*, 39(9):1061–1084, 2020. 2

[32] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963. 4

[33] Sven Middelberg, Torsten Sattler, Ole Untzelmann, and Leif Kobbelt. Scalable 6-dof localization on mobile devices. In *European conference on computer vision*, pages 268–283. Springer, 2014. 2

[34] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015. 2

[35] Raúl Mur-Artal and Juan D Tardós. Visual-inertial monocular slam with map reuse. *IEEE Robotics and Automation Letters*, 2(2):796–803, 2017. 2

[36] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE international conference on computer vision*, pages 3456–3465, 2017. 1, 2

[37] Krishna Regmi and Mubarak Shah. Bridging the domain gap for ground-to-aerial image matching. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 1, 2

[38] Paul-Edouard Sarlin, Ajaykumar Unagar, Mans Larsson, Hugo Germain, Carl Toft, Viktor Larsson, Marc Pollefeys, Vincent Lepetit, Lars Hammarstrand, Fredrik Kahl, et al. Back to the feature: Learning robust camera localization from pixels to pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3247–3257, 2021. 2

[39] Torsten Sattler, Michal Havlena, Filip Radenovic, Konrad Schindler, and Marc Pollefeys. Hyperpoints and fine vocabularies for large-scale location recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2102–2110, 2015. 2

[40] Torsten Sattler, Michal Havlena, Konrad Schindler, and Marc Pollefeys. Large-scale location recognition and the geometric burstiness problem. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1582–1590, 2016. 1, 2

[41] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1744–1756, 2016. 2

[42] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8601–8610, 2018. 2

[43] Torsten Sattler, Akihiko Torii, Josef Sivic, Marc Pollefeys, Hajime Taira, Masatoshi Okutomi, and Tomas Pajdla. Are large-scale 3d models really necessary for accurate visual localization? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1637–1646, 2017. 1, 2

[44] Thomas Schneider, Marcin Dymczyk, Marius Fehr, Kevin Egger, Simon Lynen, Igor Gilitschenski, and Roland Siegwart. maplab: An open framework for research in visual-inertial mapping and localization. *IEEE Robotics and Automation Letters*, 3(3):1418–1425, 2018. 2

[45] Yujiao Shi, Dylan John Campbell, Xin Yu, and Hongdong Li. Geometry-guided street-view panorama synthesis from satellite imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2

[46] Yujiao Shi, Liu Liu, Xin Yu, and Hongdong Li. Spatial-aware feature aggregation for image based cross-view geo-localization. In *Advances in Neural Information Processing Systems*, pages 10090–10100, 2019. 1, 2, 5, 6, 7

[47] Yujiao Shi, Xin Yu, Dylan Campbell, and Hongdong Li. Where am I looking at? joint location and orientation estimation by cross-view matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4064–4072, 2020. 1, 2, 5, 6, 7

[48] Yujiao Shi, Xin Yu, Liu Liu, Dylan John Campbell, Piotr Koniusz, and Hongdong Li. Accurate 3-dof camera geo-localization via ground-to-satellite image matching. *arXiv preprint arXiv:2203.14148*, 2022. 2

[49] Yujiao Shi, Xin Yu, Liu Liu, Tong Zhang, and Hongdong Li. Optimal feature transport for cross-view image geo-localization. In *AAAI*, pages 11990–11997, 2020. 1, 2, 5, 6

[50] Erik Stenborg, Torsten Sattler, and Lars Hammarstrand. Using image sequences for long-term visual localization. In *2020 International Conference on 3D Vision (3DV)*, pages 938–948. IEEE, 2020. 2

[51] Bin Sun, Chen Chen, Yingying Zhu, and Jianmin Jiang. Geocapsnet: Ground to aerial view image geo-localization using capsule network. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 742–747. IEEE, 2019. 1, 2

[52] Linus Svärm, Olof Enqvist, Fredrik Kahl, and Magnus Oskarsson. City-scale localization for cameras with known vertical direction. *IEEE transactions on pattern analysis and machine intelligence*, 39(7):1455–1461, 2016. 2

[53] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense

matching and view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7199–7209, 2018. 2

[54] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 8

[55] Yicong Tian, Chen Chen, and Mubarak Shah. Cross-view image matching for geo-localization in urban environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3608–3616, 2017. 1, 2

[56] Carl Toft, Will Maddern, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, et al. Long-term visual localization revisited. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[57] Aysim Toker, Qunjie Zhou, Maxim Maximov, and Laura Leal-Taixé. Coming down to earth: Satellite-to-street view synthesis for geo-localization. *CVPR*, 2021. 1, 2, 6, 7

[58] Akihiko Torii, Relja Arandjelovic, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 place recognition by view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1808–1817, 2015. 1, 2

[59] Jonathan Ventura, Clemens Arth, Gerhard Reitmayr, and Dieter Schmalstieg. Global localization from monocular slam on a mobile phone. *IEEE transactions on visualization and computer graphics*, 20(4):531–539, 2014. 2

[60] Nam N Vo and James Hays. Localizing and orienting street views using overhead imagery. In *European Conference on Computer Vision*, pages 494–509. Springer, 2016. 1, 2

[61] Lukas Von Stumberg, Patrick Wenzel, Nan Yang, and Daniel Cremers. Lm-reloc: Levenberg-marquardt based direct visual relocalization. In *2020 International Conference on 3D Vision (3DV)*, pages 968–977. IEEE, 2020. 2

[62] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf--: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 8

[63] Scott Workman and Nathan Jacobs. On the location dependence of convolutional neural network features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 70–78, 2015. 1, 2

[64] Scott Workman, Richard Souvenir, and Nathan Jacobs. Wide-area image geolocalization with aerial reference imagery. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3961–3969, 2015. 1, 2

[65] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. inerf: Inverting neural radiance fields for pose estimation. *arXiv preprint arXiv:2012.05877*, 2020. 8

[66] Bernhard Zeisl, Torsten Sattler, and Marc Pollefeys. Camera pose voting for large-scale image-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2704–2712, 2015. 2

[67] Menghua Zhai, Zachary Bessinger, Scott Workman, and Nathan Jacobs. Predicting ground-level scene layout from aerial imagery. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 3, 2017. 1, 2, 5

[68] Yao Zhou, Guowei Wan, Shenhua Hou, Li Yu, Gang Wang, Xiaofei Rui, and Shiyu Song. Da4ad: End-to-end deep attention-based visual localization for autonomous driving. In *European Conference on Computer Vision*, pages 271–289. Springer, 2020. 2

[69] Sijie Zhu, Taojiannan Yang, and Chen Chen. Revisiting street-to-aerial view image geo-localization and orientation estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 756–765, 2021. 1, 2

[70] Sijie Zhu, Taojiannan Yang, and Chen Chen. Vigor: Cross-view image geo-localization beyond one-to-one retrieval. *CVPR*, 2021. 2, 5, 6