Boosting 3-DoF Ground-to-Satellite Camera Localization Accuracy via Geometry-Guided Cross-View Transformer

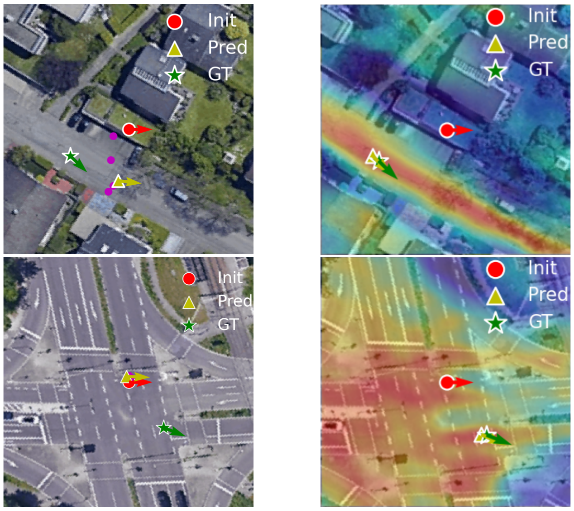
Yujiao Shi[1], Fei Wu[1], Akhil Perincherry[2], Ankit Vora[2] and Hongdong Li[1]
[1]The Australian National University [2]Ford Motor Company
yujiao.shi@anu.edu.au

Abstract

Image retrieval-based cross-view localization methods often lead to very coarse camera pose estimation, due to the limited sampling density of the database satellite images. In this paper, we propose a method to increase the accuracy of a ground camera's location and orientation by estimating the relative rotation and translation between the ground-level image and its matched/retrieved satellite image. Our approach designs a geometry-guided cross-view transformer that combines the benefits of conventional geometry and learnable cross-view transformers to map the ground-view observations to an overhead view. Given the synthesized overhead view and observed satellite feature maps, we construct a neural pose optimizer with strong global information embedding ability to estimate the relative rotation between them. After aligning their rotations, we develop an uncertainty-guided spatial correlation to generate a probability map of the vehicle locations, from which the relative translation can be determined. Experimental results demonstrate that our method significantly outperforms the state-of-the-art. Notably, the likelihood of restricting the vehicle lateral pose to be within 1m of its Ground Truth (GT) value on the cross-view KITTI dataset has been improved from 35.54% *to* 76.44%*, and the likelihood of restricting the vehicle orientation to be within* 1° *of its GT value has been improved from* 19.64% *to* 99.10%.

1. Introduction

Autonomous robots, such as unmanned aerial vehicles (UAVs) and unmanned ground vehicles (UGVs), are becoming more and more popular in various fields of applications. One of the most demanding capabilities of autonomous vehicles is navigating and executing tasks autonomously in complex environments, especially in environments with poor GPS signals. This motivates recent research on vision-based localization.



(a) Joint 3-DoF pose optimization [29] (b) Ours

Figure 1. Conventional image-retrieval methods for ground-to-satellite localization provide a rough location and orientation estimation for the ground camera, denoted by the red dot and arrow in the images. This paper aims to refine this pose under the same ground-to-satellite image-matching context. Compared to joint rotation and translation *optimization* [29], which is susceptible to local minima (a), this paper introduces a new method that allows *dense search* at all possible locations. Our method produces a probability map (b) over the continuous search space of vehicle locations, thus achieving high localization accuracy.

Ground-to-satellite image-based localization is a vision-based localization task that aims to estimate the location and orientation of a ground camera by matching a ground-level image against a large satellite map. The task was originally proposed for city-scale localization and tackled by image retrieval techniques. However, image retrieval techniques can only provide a rough pose approximation of the ground camera, and the sample density of the database image always dictates the estimated pose accuracy.

Recent works have explored increasing the localization accuracy by estimating a relative translation and rotation between the ground and its matching satellite images. Existing works have tried to regress the relative translation by

MLPs [50], or further split the retrieved satellite image to a $N \times N$ grid and match the ground image against the grid to improve the localization accuracy [43]. However, both of them cannot estimate the orientation of ground cameras. To estimate a 3-DoF camera pose (location and orientation), a deep cross-view optimization scheme has been proposed [29]. Nonetheless, optimization-based methods are highly susceptible to local minima, as shown in Fig. 1 (a).

To avoid this issue, this paper presents a new framework that allows searching for vehicle locations densely over the entire solution space. It estimates the orientation and location of the ground camera sequentially and is designed based on an observation that neural networks behave differently to rotations and translations on input signals.

Specifically, neural network outputs tend to magnify the rotation difference on input signals, making it a desired choice for rotation estimation. Thus, we propose a neural network-based pose optimizer. It works with an overhead view feature synthesis module, which synthesizes an overhead view feature map from the query ground view image, to estimate the relative rotation between the ground and satellite image. However, due to feature aggregation layers (*e.g*., pooling), a small translation difference in input signals might be absorbed in high-level deep features. This makes the estimated translation by an optimizer constructed by neural networks inaccurate.

On the other hand, when the orientation of the synthesized overhead view feature map from the ground view image has been aligned with the satellite image, the vehicle location can be obtained by performing a spatial correlation between them. The spatial correlation generates a probability map of vehicle locations over the entire search space, as shown in Fig. 1 (b). This allows a dense and exhaustive search for vehicle locations.

Our overhead view feature synthesis module is designed as a geometry-guided cross-view transformer. It explicitly embeds the deterministic geometric correspondences to the learnable cross-view transformers. Compared to the ground plane homography projection in [29], our method handles the height ambiguity of scene objects and the ground camera's slight tilt and roll angle change.

Our contributions are summarized as follows:

- a rotation and translation decoupled cross-view camera localization framework, which achieves state-of-the-art performance on four widely used benchmarks;
- a neural pose optimizer for rotation estimation, which produces highly-accurate rotation estimation results;
- a dense search mechanism for translation estimation, which computes a possibility map of vehicle locations over the entire search space;
- a geometry-guided cross-view transformer for ground-to-overhead view feature synthesis, which combines the wisdom of deterministic geometric and data-driven learnable correspondences.

2. Related Works

Satellite image-based localization. Satellite image-based localization aims to estimate the location and orientation of a ground sensor mounted on a robot or a vehicle by a large satellite image. Various works have tried localizing lidar [40, 21, 6] or radar [36, 37] points on satellite imagery. However, equipping a lidar or radar sensor on a robot is usually expensive. In contrast, cameras provide a cheaper option than lidar and radar sensors. Thus, using ground-to-satellite image matching for localization has recently attracted tremendous attention.

Ground-to-satellite image-based localization was initially proposed for city-scale localization. The task is to retrieve the most similar satellite image from a database to determine the query camera location. Conventional works have focused on designing powerful handcrafted features to match the cross-view images [3, 16, 22]. To handle significant cross-view differences, recent deep metric learning techniques provide a powerful alternative for the cross-view image matching task. Researchers have devoted themselves to designing powerful networks [41, 42, 39, 2, 44, 48], learning orientation invariant or equivariant descriptors [10, 17, 34, 31, 49], and bridging the cross-view domain gaps [45, 25, 30, 32, 38]. However, image retrieval-based methods suffer from poor localization accuracy as they approximate the GPS of retrieved satellite image as the query camera location.

Recently, researchers have demonstrated that it is possible to accurately localize which pixel on the satellite image corresponds to the query camera location. Zhu *et al*. [50] employ BlackBox MLPs to regress the relative location coordinates. Xia *et al*. [43] propose a patch-matching method to estimate the uncertainty of the ground vehicle location on a satellite image. However, the two works cannot estimate the camera's orientation with respect to the satellite image. Considering the large search space of 3-DoF camera pose, Shi and Li [29] design a deep optimization mechanism to update camera pose iteratively.

There are also two contemporary works. Lentsch *et al*. [13] propose to generate a number of candidate poses and their corresponding masks on the satellite image. The satellite features selected by these masks are matched to the query ground image to determine its pose. Instead of sampling discretized poses, our method produces continuous rotation estimates, and our translation is searched uniformly over the entire search space without being affected by the sample randomness. Fervers *et al*. [8] shares the

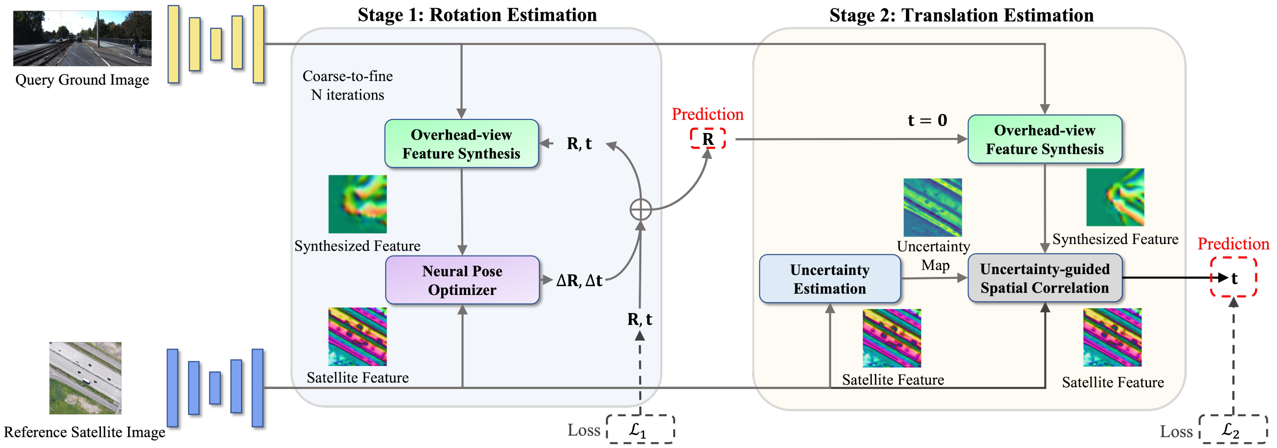


Figure 2. An overview of the proposed method. (Stage 1:) We design an overhead-view feature synthesis module to map ground-view image features to the overhead view, according to a relative rotation $\mathbf{R}$ and translation $\mathbf{t}$. Taking input as the feature differences between the synthesized and observed overhead view features, our proposed neural pose optimizer updates the relative transformation from coarse to finer feature levels. (Stage 2:) Then, we re-synthesize an overhead view feature map according to the final estimated $\mathbf{R}$ and zero translation, and use it as a sliding window to compute its spatial correlation with the observed satellite feature map. We also estimate an uncertainty map from the satellite semantic features. The uncertainty map is encoded in the spatial correlation process to exclude impossible camera locations, *e.g.*, building and tree areas. The pixel coordinate with the maximum correlation result determines the query camera location. Our neural optimizer's final estimated $\mathbf{R}$ (from stage 1) is regarded as the camera orientation.

same insight as us on translation estimation. Compared to their method, we encode an uncertainty map in the spatial correlation for translation estimation. The uncertainty map excludes improbable vehicle locations, *e.g.*, areas indicated by buildings or tree canopy. Furthermore, their rotations are sampled at discretized values, while our method estimates continuous rotations.

Decoupling rotation and translation has also been explored in other tasks, *e.g.*, absolute pose regression [11], Visual Odometry (VO) [12], and 6-DoF object pose estimation [5]. This paper shows that developing different rotation and translation estimation strategies also facilitates the overall performance of ground-to-satellite camera localization. Our proposed method tackles the unique challenges of this task and will be illustrated in detail in the next section.

Cross-view image synthesis. The cross-view image synthesis task aims to synthesize an image from one viewpoint to another viewpoint. This task was first proposed by Regmi and Borji [23], where a conditional GAN is used to learn the transformations between the two images. Since then, different methods have been proposed to improve the performance [24, 35, 19, 28, 14]. Recent research shows that cross-view image localization and synthesis tasks can complement each other, improving their performance [25, 38]. In this work, we use cross-view feature synthesis instead of image synthesis to facilitate cross-view localization performance.

Cross-view transformers. Overhead view, also known as Bird's eye view (BEV), representation learning [46, 26, 47, 4, 15, 27] has also been shown to be useful in many autonomous driving tasks, such as map-view semantic segmentation [26, 47], 3D lane line detection [4], and 3D object detection [15]. A common strategy is to learn an implicit overhead view embedding and then use the learned embedding as query features to collect ground view features and update the overhead view feature maps. Cross-view correspondences are learned implicitly from training. Instead of learning cross-view correspondences implicitly, this paper proposes a geometry-guided cross-view transformer, which explicitly encodes the geometric correspondences to the learnable transformer. This eases the burden of neural networks and significantly reduces training time.

3. Boosting 3-DoF Camera Pose Accuracy

Given a coarse location and rotation estimate of a ground camera, this paper aims to refine this camera pose by ground-to-satellite image matching. The coarse camera pose estimates can be provided by city-scale ground-to-satellite image retrieval, VO, SLAM, or imprecise sensors (*e.g.*, consumer-level GPS and compass).

3.1. Method Overview

Humans usually determine a ground camera's pose relative to a satellite image by first mentally hallucinating overhead view appearances of the observed scenes and then comparing them with the satellite map. Inspired by this, we propose a geometry-guided cross-view transformer for ground-to-overhead view feature synthesis to mimic the "hallucination" process. Then, a neural pose optimizer is

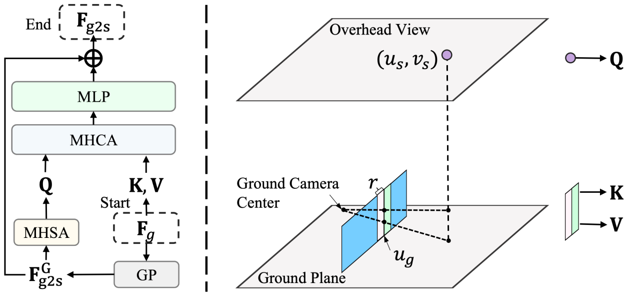


Figure 3. Ground-to-overhead feature synthesis. (Left:) A geometry projection (GP) module is first adopted to project ground image features to an overhead view by exploiting the ground plane homography. Then, a multi-head self-attention (MHSA) module is applied to the GP-projected features $\mathbf{F}^G_{g2s}$ to make each element aware of its contextual information. Next, a multi-head cross-attention (MHCA) module is proposed to further collect features from the ground-view images and update the overhead view feature representation. (Right:) For each overhead view feature map pixel ($\mathbf{Q}$), we find its corresponding column in the ground view feature map and the neighboring columns with a radius of r, constructing a feature candidate pool ($\mathbf{K}$, $\mathbf{V}$) to be used in MHCA.

constructed to perform the "comparison" behavior.

The neural optimizer constructed by neural networks can produce accurate and reliable rotation estimations, because a rotation on the input can be magnified on the network output. However, a small translation on the optimizer input is likely to be absorbed by high-level deep features inside the optimizer, leading to inaccurately estimated translations. Nonetheless, we have a more effective method for translation estimation.

When the ground camera's orientation is estimated, its location can be computed by a spatial correlation between the synthesized overhead view feature map according to the estimated rotation and the observed satellite feature map. The spatial correlation densely searches every possible location on the satellite image, avoiding local minima problems. We also encode an uncertainty map in the spatial correlation process to exclude impossible camera locations (e.g., areas indicated by buildings and tree canopies). The pixel position corresponding to the maximum correlation result indicates the most likely ground camera's location.

Fig. 2 provides an overview of our proposed method. Next, we provide technical details for each component.

3.2. Overhead-view feature synthesis

Given a ground image, we synthesize an overhead view feature map from the ground view image according to a relative rotation and translation. Our overhead-view feature synthesis module combines deterministic geometric correspondences and learnable cross-view transformers.

Geometry correspondences. We use azimuth angle θ to parameterize the ground camera's relative rotation $\mathbf{R}$ to the satellite image. We set the tilt and roll angles as zero since satellite images cannot provide reference for them and they are typically small in autonomous driving scenarios. The relative camera translation is parameterized as $\mathbf{t} = [t_x, 0, t_z]^T$, where t_x and t_z denote the translation along the latitude and longitude directions in the geographical coordinates, corresponding to the v and u directions of the satellite image coordinates, respectively. The relative translation t_y between a ground and a satellite camera along the vertical direction is infinite. Thus we do not consider it in our task by setting it as zero. The mapping from an overhead view pixel (u_s, v_s) to a ground view image pixel (u_g, v_g) is derived as:

$$\begin{bmatrix} u_g \\ v_g \end{bmatrix} = \begin{bmatrix} f_x \frac{[(v_s - v_s^0) + t_x] \cos\theta - [(u_s - u_s^0) + t_z] \sin\theta}{[(v_s - v_s^0) + t_x] \sin\theta + [(u_s - u_s^0) + t_z] \cos\theta} + u_g^0 \\ f_y \frac{h}{\alpha\{[(v_s - v_s^0) + t_x] \sin\theta + [(u_s - u_s^0) + t_z] \cos\theta\}} + v_g^0 \end{bmatrix}, \tag{1}$$

where (u_s^0, v_s^0) and (u_g^0, v_g^0) denote the overhead view feature map and the ground-view image centers, respectively, α is the ground resolution of the overhead view feature map, f_x an f_y denote the ground camera focal length along u and v directions, respectively, h is the height of pixel (u_s, v_s) above the ground plane.

It can be seen that the value of u_g can always be determined whenever the height of the corresponding pixel is given or not, and the height only affects the value of v_g.

Geometry-guided cross-view transformer. Next, we leverage cross-view transformers to handle the ambiguity of v_g. Commonly designed cross-view transformers [26, 47, 15, 4] initialize a latent overhead view embedding that is shared by different overhead view maps. The embedding functions as "query" features to collect ground-view image features and is learned by statistically data-driven training, enabling the network to collect information from ground-view images automatically.

In the cross-view localization task, the appearance of overhead view features at different geographical locations differs, and these differences are essential to localization accuracy. To address this issue, we design a geometry-guided cross-view transformer with a scene-specific "query" embedding initialization strategy, as shown in Fig. 3 (left).

First, the geometry projection (GP) module projects the ground-view features to the overhead view by exploiting the ground plane homography. This establishes authentic cross-view correspondences for scene contents on the ground plane. We then apply a multi-head self-attention (MHSA) block to the geometry projected features, making them aware of their context information, especially for the scene contents above the ground plane. The output of MHSA is used as our overhead view "query" embedding, which is scene-specific.

We have derived from Eq. (1) that the column index u_g of a satellite pixel on the ground view image can always be determined. It implies that its corresponding ground-view image pixel lies on a column on the ground-view image. When there is a slight tilt or pitch angle change of the ground camera during driving, the value of u_g will be slightly off (the value of v_g remains unknown). With this observation, we retrieve the entire column indexed by u_g and its neighbor columns with a radius of r in the ground-view image, as shown in Fig. 3 (right), which is to construct the candidate feature pool ("key" and "value") for this satellite pixel. We then use the strength of transformers to update the overhead view feature map by multi-head cross-attention (MHCA):

$$\text{MHCA}\left(\mathbf{F}_{g2s}^{G}, \mathbf{F}_g\right) = \text{Softmax}\left(\mathbf{Q}\mathbf{K}^T\right)\mathbf{V}, \tag{2}$$

$$\mathbf{Q} = \mathcal{Q}\left(\text{MHSA}\left(\mathbf{F}_{g2s}^{G}\right)\right)\ \mathbf{K} = \mathcal{K}\left(\mathbf{F}_g^L\right)\ \mathbf{V} = \mathcal{V}\left(\mathbf{F}_g^L\right), \tag{3}$$

where $\mathbf{F}_g$ denote the ground-view image features map, and $\mathbf{F}_g^L$ indicate the retrieved local region for each satellite pixel from $\mathbf{F}_g$ illustrated in Fig. 3, $\mathcal{Q}(\cdot), \mathcal{K}(\cdot), \mathcal{V}(\cdot)$ are linear mapping layers. We then feed the updated features by MHCA to an MLP layer $\mathcal{M}(\cdot)$ with a skip connection, obtaining the final synthesized overhead view feature map:

$$\mathbf{F}_{g2s} = \mathbf{F}_{g2s}^{G} + \mathcal{M}\left(\text{MHCA}\left(\mathbf{F}_{g2s}^{G}, \mathbf{F}_g\right)\right) \tag{4}$$

For memory efficiency, this cross-view transformer is only applied to the coarsest feature levels (*i.e.*, $\frac{1}{8}$ of the original image size). This also guarantees a large receptive field for cross-attention. We adopt a decoder to recover finer details of the overhead view map at higher resolutions.

3.3. Neural pose optimizer

The overhead view feature synthesis module has aligned the ground view observations and the satellite image in the same (overhead) domain. We next design a neural pose optimizer to estimate the relative pose between them, especially the relative rotation. Our neural optimizer takes input as the differences between the synthesized and observed feature maps $\mathbf{F}_{g2s} - \mathbf{F}_s$, where $\mathbf{F}_s$ denote the observed satellite image features, and outputs a relative pose update based on the current pose estimation. It is constructed by two swin transformer layers [18] and two MLP layers. The swin transformer layers are to increase the global information extraction ability of the neural optimizer to avoid local minima, especially for the estimated rotations. Here, we make the optimizer update the translation as well. The reason is to encourage the optimizer to find a translation, based on which the rotation can be easily and correctly estimated. The neural pose optimizer is applied from coarse to finer feature levels and then repeated. This allows fine-tuning around a potential global minimum and jumping out of local minima.

3.4. Uncertainty-guided spatial correlation

After the ground camera's orientation has been estimated, we re-synthesize an overhead view feature map from the ground view observation according to the estimated orientation and zero translation. The zero translation here is to make the ground camera's location correspond to the center of the synthesized overhead view feature map. Then, we compute the relative translation between the query camera location and the satellite image center by a spatial correlation between the synthesized overhead-view features and the observed satellite image features. This is shown in the numerator of Eq. (5). We also consider that the satellite semantics themselves encode a likelihood of the ground vehicles' location, such that a car is likely on the road but unlikely on top of buildings. To account for this, we estimate an uncertainty map from the satellite image features using a set of CNNs and develop an uncertainty-guided similarity-matching scheme:

$$\mathbf{P} = \left(\mathbf{F}_s \star \mathbf{F}_{g2s}\right)(u_s, v_s) / \mathbf{U}(u_s, v_s), \tag{5}$$

where $\star$ denotes the normalized cross-correlation, $\mathbf{U}$ is the uncertainty map estimated from satellite semantics (features). The value of the uncertainty is within the range of $(0, 1)$. The higher the value, the lower the possibility of the vehicle being at the corresponding location.

The correlation results, $\mathbf{P}$, indicate the final computed possibility of the query camera being at each of the satellite image pixels. A higher value implies a large probability of the query camera being at the corresponding location. We regard the pixel location with the highest correlation value as the query camera location.

3.5. Training objective

We use ground truth pose as supervision for the neural optimizer output:

$$\mathcal{L}_1 = \sum_n \sum_l (\|\theta_n^l - \theta^*\|_1 + \|{t_x}_n^l - {t_x}^*\|_1) + \|{t_z}_n^l - {t_z}^*\|_1), \tag{6}$$

where $l = [1, 2, 3]$ and $n = [1, 2]$ index the feature levels and the number of iterations, respectively, $\theta_n^l, {t_x}_n^l, {t_z}_n^l$ denote the predicted camera pose (rotation and translation) by the neural optimizer at feature level l and iteration number n, $\theta^*, {t_x}^*, {t_z}^*$ represent the ground truth (GT) camera pose. The predicted translation by the neural optimizer is not taken as our final output, but we still apply supervision for them here. This is to encourage the gradients of trainable parameters in the neural optimizer to be updated smoothly and in the correct directions.

We apply a triplet loss to the spatial correlation results for translation estimation. We aim to maximize the possibility at GT (positive) camera location while minimizing

Table 1. Comparison results on KITTI and Ford Multi-AV with aligned orientation. "*" indicates fine-grained image retrieval methods.

	Lateral		Longitudinal		Lateral		Longitudinal		Lateral		Longitudinal		Lateral		Longitudinal	
	$d=1$	$d=3$	$d=1$	$d=3$	$d=1$	$d=3$	$d=1$	$d=3$	$d=1$	$d=3$	$d=1$	$d=3$	$d=1$	$d=3$	$d=1$	$d=3$
	KITTI - Test1				KITTI - Test2				Ford - Log1				Ford - Log2			
CVM-NET* [10]	3.87	12.38	3.81	11.16	3.87	12.38	3.81	11.16	9.62	25.33	4.10	12.29	10.33	29.17	4.11	12.80
CVFT* [32]	13.04	36.84	4.06	11.50	12.19	34.23	3.75	10.36	15.14	40.71	4.38	12.19	15.00	40.94	4.70	14.76
SAFA* [30]	13.20	36.76	4.21	12.51	13.35	36.93	4.39	11.99	11.33	31.62	4.33	13.05	15.16	42.45	4.72	13.71
Polar-SAFA* [30]	13.44	37.11	4.96	13.12	13.76	38.42	3.92	11.27	13.38	37.19	3.86	10.81	14.57	40.41	4.35	12.74
DSM* [31]	13.86	38.17	4.43	11.93	13.54	37.40	4.00	11.71	12.24	33.14	4.48	12.10	12.18	33.67	4.45	12.50
VIGOR* [50]	14.18	38.38	4.61	13.25	13.07	36.40	4.48	12.20	11.76	35.43	6.24	17.57	19.48	62.38	5.04	15.88
L2LTR* [44]	15.19	41.93	5.33	14.37	14.65	40.36	4.69	12.40	14.19	38.10	4.71	13.29	13.52	38.07	4.45	12.96
TansGeo* [48]	15.74	43.15	5.04	14.44	14.85	41.37	4.31	12.64	14.57	39.76	4.10	13.86	13.57	39.76	4.10	13.86
LM [29]	52.66	82.14	4.35	14.95	37.63	69.07	4.96	14.66	64.05	83.05	11.38	22.57	47.49	79.18	5.31	15.59
CVML [43]	64.27	83.12	34.77	65.04	19.54	49.83	10.47	25.86	53.00	83.67	6.19	18.62	22.16	37.54	5.31	15.56
Ours	**93.85**	**98.44**	**52.40**	**79.75**	**55.28**	**85.73**	**17.97**	**39.49**	**80.76**	**95.90**	**28.48**	**38.57**	**78.19**	**89.21**	**22.86**	**40.43**

Table 2. Comparison results on Oxford RobotCar with aligned orientation. The lower, the better.

	Test1		Test2		Test3		Overall	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median
CVML [43]	**1.66**	1.29	**2.28**	1.55	**2.26**	1.51	**2.07**	1.44
Ours	2.40	**0.91**	3.10	**1.13**	2.86	**1.05**	2.79	**1.02**

Table 3. Comparison results on VIGOR with aligned orientation.

Algorithms	Same-area		Cross-area	
	Mean	Median	Mean	Median
CVML [43]	6.94	3.64	9.05	5.14
SliceMatch [13]	5.18	2.58	5.53	2.55
Ours	**4.12**	**1.34**	**5.16**	**1.40**

that at other (negative) locations:

$$\mathcal{L}_2 = \frac{1}{|\mathbf{P}|} \sum_{(u_s, v_s)} log(1 + e^{\gamma(\mathbf{P}(u_s^*, v_s^*) - \mathbf{P}(u_s, v_s))}), \quad (7)$$

where (u_s^*, v_s^*) indicates the pixel corresponding to the GT camera location, (u_s, v_s) denotes other pixels, $|\mathbf{P}|$ is the total number of pixels in $\mathbf{P}$, and γ is set to 10. We should note that we do not provide explicit supervision for the uncertainty maps in Eq. (5). Rather, they are learned statistically and implicitly from the translational pose training loss represented by Eq. (7).

We follow Kendall and Roberto [11] to balance the two training loss items:

$$\mathcal{L} = \mathcal{L}_1 e^{-\lambda_1} + \lambda_1 + \mathcal{L}_2 e^{-\lambda_2} + \lambda_2 \quad (8)$$

where λ_1 is initialized as -5 and λ_2 as -3. They are learned and adjusted dynamically during training.

4. Experiments

Dataset and evaluation metrics. We evaluate the performance of our method and compare it with state-of-the-art on several datasets, including the cross-view KITTI [9, 29] and Ford Multi-AV [1, 29] datasets, the Oxford RobotCar dataset [20, 43] as well as the VIGOR dataset [50].

The cross-view **KITTI** dataset consists of one training set and two test sets. Test1 contains images sampled from the same region as the training set, while Test2 contains images from a different region. The cross-view **Ford Multi-AV** dataset [1] includes six subsets (Log1-Log6), each captured on two different dates along different trajectories. Images from one date are used for training, while those from the other are used for testing. We follow Shi and Li [29] for the evaluation of these two datasets. Specifically, a query image is considered correctly localized in a direction if its estimated translation along that direction is within d meters of the ground truth translation. The percentage of correctly localized query images in that direction is recorded. We also report the percentage of images with correctly estimated orientation, when the estimated orientation of the query image is within θ degrees of its ground truth orientation. We present results for the first two logs of the Ford dataset in the main paper and include results for the remaining logs in the supplementary material.

The cross-view **Oxford RobotCar** dataset includes one training set, one validation set, and three test sets. The test sets consist of images from three traversals captured at different dates than those in the training set. The **VIGOR** dataset contains cross-view images from four cities in the USA, Chicago, New York, San Francisco, and Seattle. The dataset contains two evaluation scenarios: same-area and cross-area evaluation. We use the rectified labels by SliceMatch [13] and follow its evaluation protocol by only using the fully positive satellite images. Both the Oxford RobotCar and the VIGOR datasets assume orientation alignment and do not provide a test set with unknown orientations. Therefore, we only test location estimation with aligned orientation. We follow the evaluation protocol outlined in its original paper [43], reporting mean and median distances between predicted and ground truth locations. We perform joint location and orientation estimation on the KITTI and

Ford Multi-AV datasets, as they provide official test sets for this challenging scenario.

Implementation details. Our network backbone uses a U-Net architecture with the encoder being a VGG16 [33] pretrained on ImageNet [7]. The decoder weights are randomly initialized. The satellite branch has two output heads: one for satellite features and the other for uncertainty maps. The satellite image resolution for all datasets is set to 512×512, following the original settings [29, 43]. The ground image resolution is 256×1024 for the cross-view KITTI and Ford Multi-AV datasets, and 154×231 for the cross-view Oxford RobotCar dataset. The local region radius in MHCA is set to 1. The learning rate starts from 10^{-4} and gradually decreases to 10^{-5}. The network is trained for five epochs with a batch size of 3. Using an RTX 3090 GPU, the inference time for each image is 280ms, significantly faster than Deep LM [29], which requires 500ms. Code is available at `https://github.com/shiyujiao/Boosting3DoFAccuracy.git`.

4.1. Comparison with the state-of-the-art

We compare our method with the state-of-the-art, including fine-grained image retrieval [10, 32, 30, 31, 50], deep LM optimization [29] and cross-view metric learning (CVML) [43]. For the fine-grained image retrieval, we split the satellite image into $N \times N$ small patches and retrieve the most similar patch. The retrieved patch center is regarded as the query camera location. Results are from paper [29] or re-evaluated by the author-provided models and codes. All experiment settings follow previous works [29, 43].

Location estimation with given orientation. Since most of the existing cross-view localization methods have an assumption of aligned orientation and only target location estimation, we first compare the performance of our method with them on localizing orientation-aligned images. The results on KITTI and Ford Multi-AV datasets are presented in Tab. 1. It can be seen that all fine-grained image retrieval-based methods achieve inferior results. This is because the images in the fine-grained database are too similar to disambiguate. Benefiting from the specific design for the accurate cross-view localization task, the Deep LM [29] and CVML [43] significantly improve the results. Nonetheless, our method increases the performance considerably, *e.g.* from 64.27% to 93.85% on Test1 of KITTI.

The comparison results on the Oxford RobotCar and VIGOR datasets are presented in Tab. 2 and Tab. 3. We also compare with SliceMatch [13], a contemporary work with our submission, on the VIGOR dataset. The results are from its original paper. The comparison reveals that our method achieves the best performance on most scenarios. Our method achieves a better median error while a worse mean error than CVML on the Oxford RobotCar dataset, with a similar pattern mirrored on the VIGOR dataset, where our improvement on the median error over CVML is more significant than on the mean error. The reason is that our method utilizes similarity matching instead of network regression for location estimation.

When the scene is diverse and no symmetry exists, the location computed by similarity matching is closer to the GT value than regressed by networks, leading to a smaller median error. However, in rare cases where scene symmetry exists at “different and distant” locations, the location computed by similarity matching can be far from the GT value. This increases the mean error of our method. In contrast, the network regression-based method (CVML) tends to resemble the GT values, resulting in a smaller mean error, especially in cases when training and testing images are from the same area (RobotCar). The VIGOR dataset contains images from four cities (Vs. one city in RobotCar), and the number of testing images in VIGOR is around $10\times$ larger than those in RobotCar. Thus, the VIGOR dataset is more diverse, and our method achieves consistently better results than CVML. More importantly, our improvement over CVML is more considerable on the cross-area evaluation than the same-area evaluation on VIGOR, and our generalization from Test1 to Test2 on the KITTI dataset is also better than CVML.

Joint location and orientation estimation. Next, we evaluate the performance of different methods on joint location and orientation estimation. Among the comparison algorithms, only DSM [31] and Deep LM [29] are designed for this purpose. The results are presented in Tab. 4. It can be seen that our method achieves the best performance among the comparison algorithms on all the test sets. Remarkably, we obtain an exceptional 99% likelihood for restricting the estimated rotation to be 1° to its GT value for both test sets on KITTI, improving from around 19.64% on Test1 and 46.82% on Test2. This should be attributed to the powerfulness of our neural optimizer on rotation estimation. Benefiting from the high rotation estimation accuracy and the dense search scheme, we nearly double the results on lateral pose estimation with $d = 1$ for almost all the test sets. Fig. 4 shows some examples of our localization results.

4.2. Model analysis

G2S feature synthesis alternatives. We first compare different ground-to-satellite feature synthesis alternatives, including pure geometry projection [29] and Persformer [4] (a geometry-guided deformable cross-view transformer). Since our method leverages the merits of both conventional geometry and learnable cross-view transformers, our method achieves the best performance compared to the different alternatives, as indicated in the first part of Tab. 5.

Effectiveness of uncertainty maps. Next, we conduct experiments to demonstrate the effectiveness of uncertainty maps estimated from satellite images. Fig. 5 shows some

Table 4. Performance comparison on KITTI and Ford Multi-AV with 20° orientation noise.

	Lateral			Longitudinal			Azimuth			Lateral			Longitudinal			Azimuth		
	$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$	$d=1$	$d=3$	$d=5$	$d=1$	$d=3$	$d=5$	$\theta=1$	$\theta=3$	$\theta=5$
	KITTI - Test1									KITTI - Test2								
DSM* [31]	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
LM [29]	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
SliceMatch [13]	49.49	–	98.52	15.19	–	57.35	13.41	–	64.17	32.43	–	86.44	8.30	–	35.57	46.82	–	46.82
Ours	**76.44**	**96.34**	**98.89**	**23.54**	**50.57**	**62.18**	**99.10**	**100.00**	**100.00**	**57.72**	**86.77**	**91.16**	**14.15**	**34.59**	**45.00**	**98.98**	**100.00**	**100.00**
	Ford - Log1									Ford - Log2								
DSM* [31]	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
LM [29]	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
Ours	**70.00**	**93.10**	**95.52**	**17.10**	**27.95**	**29.76**	**59.38**	**90.57**	**95.24**	**60.26**	**76.58**	**95.12**	**20.77**	**37.67**	**39.39**	**62.68**	**93.78**	**98.77**

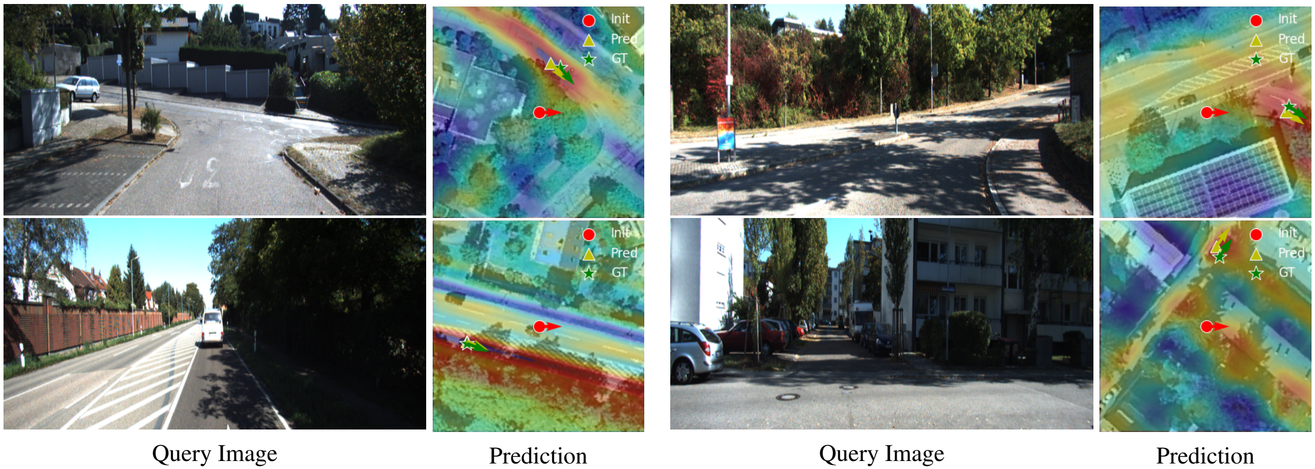


Figure 4. Localization results visualization. The correlation results (vehicle location possibility maps) overlay the satellite images

Table 5. Ablation study results on KITTI with 20° orientation noise.

		Lateral		Longitudinal		Azimuth		Lateral		Longitudinal		Azimuth	
		$d=1$	$d=3$	$d=1$	$d=3$	$\theta=1$	$\theta=3$	$d=1$	$d=3$	$d=1$	$d=3$	$\theta=1$	$\theta=3$
		Test1						Test2					
G2S Synthesis Alternatives	Geometry Projection	75.96	95.68	22.55	46.67	97.59	**100.00**	54.80	85.68	14.00	30.99	98.34	100.00
	Persformer [4]	68.04	93.85	5.51	15.77	97.98	99.97	38.86	73.56	3.71	11.77	98.42	100.00
	Ours	**76.44**	**96.34**	**23.54**	**50.57**	**99.10**	**100.00**	**57.72**	**86.77**	**14.15**	**34.59**	**98.98**	**100.00**
Uncertainty	Without	70.29	94.30	18.29	40.39	84.44	99.84	54.53	85.44	12.50	29.65	84.16	99.80
	With (Ours)	**76.44**	**96.34**	**23.54**	**50.57**	**99.10**	**100.00**	**57.72**	**86.77**	**14.15**	**34.59**	**98.98**	**100.00**

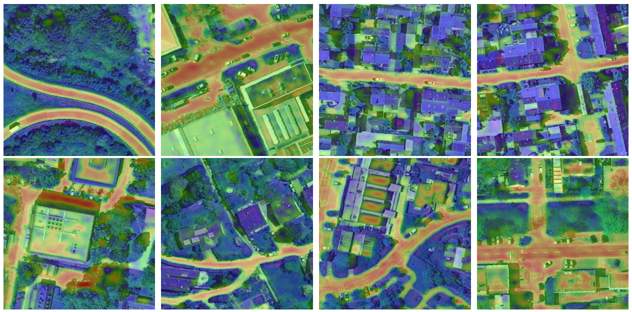


Figure 5. Learned inverse uncertainty maps from satellite features, where road regions are highlighted with small uncertainty.

examples of the inverse uncertainty (confidence) maps with road regions highlighted. These regions represent the likelihood of the presence of a vehicle. The last two rows in Tab. 5 present the performance of our method with or without uncertainty maps in translation estimation. It can be seen that even without the uncertainty maps, our method can still achieve promising localization results. With the uncertainty maps, the performance is further improved.

Due to space limits, we discuss the comparison between ground-to-satellite projection and satellite-to-ground projection, the sensitivity of our method to different initial poses, the iteration number choice of the neural pose optimizer, and additional ablation study results in the supplementary material.

5. Discussion and Conclusion

Various methods can provide a coarse estimation of a camera's pose at ground level, such as city-scale ground-to-satellite image retrieval, VO, SLAM, or sensors (noisy

GPS, compass, wheel encoder, etc.). Given a coarse rotation and translation estimate, this paper has presented a new framework to improve the ground camera's pose accuracy by ground-to-satellite image matching. This task has various applications, such as reducing costs in autonomous driving and field robotics by using a cheaper GPS device. It can also function as a new loop closure method for SLAM and VO techniques.

Our approach consists of a geometry-guided cross-view transformer for ground-to-overhead feature synthesis, a neural pose optimizer for rotation estimation, and an uncertainty-guided spatial correlation for translation estimation. It significantly outperforms the current state-of-the-art in various challenging localization scenarios.

The single query image setting brings some limitations for this task. For example, the longitudinal pose estimation accuracy is not as accurate as lateral poses due to the monotonous scene appearance along driving directions. Moreover, the field of view of a single camera is limited. When the rotation and translation ambiguity (search space) is too large, scene content captured by a single camera can be similar at different poses, leading to inaccurate localization performance. However, these limitations can be potentially solved by using a multi-camera system or a continuous ground-view video for localization. These multi-frames will increase the informativeness of the query place and thus the localization accuracy. We will investigate these possibilities in the future.

6. Acknowledgments

The research is funded in part by an ARC Discovery Grant (grant ID: DP220100800)to HL. We thank all anonymous reviewers and ACs for their constructive suggestions.

References

[1] Siddharth Agarwal, Ankit Vora, Gaurav Pandey, Wayne Williams, Helen Kourous, and James McBride. Ford multi-av seasonal dataset, 2020. 6

[2] 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. 2

[3] 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

[4] Li Chen, Chonghao Sima, Yang Li, Zehan Zheng, Jiajie Xu, Xiangwei Geng, Hongyang Li, Conghui He, Jianping Shi, Yu Qiao, et al. Persformer: 3d lane detection via perspective transformer and the openlane benchmark. *arXiv preprint arXiv:2203.11089*, 2022. 3, 4, 7, 8

[5] Wei Chen, Xi Jia, Hyung Jin Chang, Jinming Duan, Linlin Shen, and Ales Leonardis. Fs-net: Fast shape-based network for category-level 6d object pose estimation with decoupled rotation mechanism. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1581–1590, 2021. 3

[6] Lucas De Paula Veronese, Edilson de Aguiar, Rafael Correia Nascimento, Jose Guivant, Fernando A. Auat Cheein, Alberto Ferreira De Souza, and Thiago Oliveira-Santos. Re-emission and satellite aerial maps applied to vehicle localization on urban environments. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4285–4290, 2015. 2

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 7

[8] Florian Fervers, Sebastian Bullinger, Christoph Bodensteiner, Michael Arens, and Rainer Stiefelhagen. Uncertainty-aware vision-based metric cross-view geolocalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21621–21631, 2023. 2

[9] 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. 6

[10] 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. 2, 6, 7

[11] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5974–5983, 2017. 3, 6

[12] Pyojin Kim, Brian Coltin, and H Jin Kim. Low-drift visual odometry in structured environments by decoupling rotational and translational motion. In *2018 IEEE international conference on Robotics and automation (ICRA)*, pages 7247–7253. IEEE, 2018. 3

[13] Ted Lentsch, Zimin Xia, Holger Caesar, and Julian FP Kooij. Slicematch: Geometry-guided aggregation for cross-view pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17225–17234, 2023. 2, 6, 7, 8

[14] Zuoyue Li, Zhenqiang Li, Zhaopeng Cui, Rongjun Qin, Marc Pollefeys, and Martin R Oswald. Sat2vid: Street-view panoramic video synthesis from a single satellite image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12436–12445, 2021. 3

[15] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *arXiv preprint arXiv:2203.17270*, 2022. 3, 4

[16] Tsung-Yi Lin, Serge Belongie, and James Hays. Cross-view image geolocalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 891–898, 2013. 2

[17] 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. 2

[18] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 5

[19] Xiaohu Lu, Zuoyue Li, Zhaopeng Cui, Martin R Oswald, Marc Pollefeys, and Rongjun Qin. Geometry-aware satellite-to-ground image synthesis for urban areas. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 859–867, 2020. 3

[20] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017. 6

[21] Subodh Mishra, Armin Parchami, Enrique Corona, Punarjay Chakravarty, Ankit Vora, Devarth Parikh, and Gaurav Pandey. Localization of a smart infrastructure fisheye camera in a prior map for autonomous vehicles. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 5998–6004, 2022. 2

[22] Arsalan Mousavian and Jana Kosecka. Semantic image based geolocation given a map. *arXiv preprint arXiv:1609.00278*, 2016. 2

[23] Krishna Regmi and Ali Borji. Cross-view image synthesis using conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3501–3510, 2018. 3

[24] Krishna Regmi and Ali Borji. Cross-view image synthesis using geometry-guided conditional gans. *Computer Vision and Image Understanding*, 187:102788, 2019. 3

[25] 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. 2, 3

[26] Thomas Roddick and Roberto Cipolla. Predicting semantic map representations from images using pyramid occupancy networks. in 2020 ieee. In *CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 13–19, 2020. 3, 4

[27] Avishkar Saha, Oscar Mendez, Chris Russell, and Richard Bowden. Translating images into maps. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 9200–9206. IEEE, 2022. 3

[28] Yujiao Shi, Dylan Campbell, Xin Yu, and Hongdong Li. Geometry-guided street-view panorama synthesis from satellite imagery. *arXiv preprint arXiv:2103.01623*, 2021. 3

[29] Yujiao Shi and Hongdong Li. 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*, pages 17010–17020, 2022. 1, 2, 6, 7, 8

[30] 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. 2, 6, 7

[31] 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. 2, 6, 7, 8

[32] 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. 2, 6, 7

[33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 7

[34] 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. 2

[35] Hao Tang, Dan Xu, Nicu Sebe, Yanzhi Wang, Jason J Corso, and Yan Yan. Multi-channel attention selection gan with cascaded semantic guidance for cross-view image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2417–2426, 2019. 3

[36] Tim Yuqing Tang, Daniele De Martini, Dan Barnes, and Paul Newman. Rsl-net: Localising in satellite images from a radar on the ground. *IEEE Robotics and Automation Letters*, 5(2):1087–1094, 2020. 2

[37] Tim Y Tang, Daniele De Martini, Shangzhe Wu, and Paul Newman. Self-supervised learning for using overhead imagery as maps in outdoor range sensor localization. *The International Journal of Robotics Research*, 40(12-14):1488–1509, 2021. 2

[38] Aysim Toker, Qunjie Zhou, Maxim Maximov, and Laura Leal-Taixé. Coming down to earth: Satellite-to-street view synthesis for geo-localization. *CVPR*, 2021. 2, 3

[39] 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. 2

[40] Ankit Vora, Siddharth Agarwal, Gaurav Pandey, and James McBride. Aerial imagery based lidar localization for autonomous vehicles. *arXiv preprint arXiv:2003.11192*, 2020. 2

[41] 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. 2

[42] 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. 2

[43] Zimin Xia, Olaf Booij, Marco Manfredi, and Julian FP Kooij. Visual cross-view metric localization with dense uncertainty estimates. In *European Conference on Computer Vision*, pages 90–106. Springer, 2022. 2, 6, 7

[44] Hongji Yang, Xiufan Lu, and Yingying Zhu. Cross-view geo-localization with layer-to-layer transformer. *Advances in Neural Information Processing Systems*, 34:29009–29020, 2021. 2, 6

[45] 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. 2

[46] Daiming Zhang, Bin Fang, Weibin Yang, Xiaosong Luo, and Yuanyan Tang. Robust inverse perspective mapping based on vanishing point. In *Proceedings 2014 IEEE International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, pages 458–463. IEEE, 2014. 3

[47] Brady Zhou and Philipp Krähenbühl. Cross-view transformers for real-time map-view semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13760–13769, 2022. 3, 4

[48] Sijie Zhu, Mubarak Shah, and Chen Chen. Transgeo: Transformer is all you need for cross-view image geo-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1162–1171, June 2022. 2, 6

[49] 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. 2

[50] Sijie Zhu, Taojiannan Yang, and Chen Chen. Vigor: Cross-view image geo-localization beyond one-to-one retrieval. *CVPR*, 2021. 2, 6, 7