TU Delft

Delft University of Technology

Visual Cross-View Metric Localization with Dense Uncertainty Estimates

Xia, Zimin; Booij, Olaf; Manfredi, Marco; Kooij, Julian F.P.

DOI
10.1007/978-3-031-19842-7_6

Publication date
2022

Document Version
Final published version

Published in
Computer Vision – ECCV 2022

Citation (APA)
Xia, Z., Booij, O., Manfredi, M., & Kooij, J. F. P. (2022). Visual Cross-View Metric Localization with Dense Uncertainty Estimates. In S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, & T. Hassner (Eds.), *Computer Vision – ECCV 2022 : Proceedings of the 17th European Conference* (pp. 90-106). (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Vol. 13699 LNCS). Springer. https://doi.org/10.1007/978-3-031-19842-7_6

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

https://www.openaccess.nl/en/you-share-we-take-care

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

Visual Cross-View Metric Localization with Dense Uncertainty Estimates

Zimin Xia[1(✉)], Olaf Booij[2], Marco Manfredi[2], and Julian F. P. Kooij[1]

[1] Intelligent Vehicles Group, Technical University Delft, Delft, The Netherlands
{z.xia,j.f.p.kooij}@tudelft.nl
[2] TomTom, Amsterdam, The Netherlands
{olaf.booij,marco.manfredi}@tomtom.com

Abstract. This work addresses visual cross-view metric localization for outdoor robotics. Given a ground-level color image and a satellite patch that contains the local surroundings, the task is to identify the location of the ground camera within the satellite patch. Related work addressed this task for range-sensors (LiDAR, Radar), but for vision, only as a secondary regression step after an initial cross-view image retrieval step. Since the local satellite patch could also be retrieved through any rough localization prior (e.g. from GPS/GNSS, temporal filtering), we drop the image retrieval objective and focus on the metric localization only. We devise a novel network architecture with denser satellite descriptors, similarity matching at the bottleneck (rather than at the output as in image retrieval), and a dense spatial distribution as output to capture multi-modal localization ambiguities. We compare against a state-of-the-art regression baseline that uses global image descriptors. Quantitative and qualitative experimental results on the recently proposed VIGOR and the Oxford RobotCar datasets validate our design. The produced probabilities are correlated with localization accuracy, and can even be used to roughly estimate the ground camera's heading when its orientation is unknown. Overall, our method reduces the median metric localization error by 51%, 37%, and 28% compared to the state-of-the-art when generalizing respectively in the same area, across areas, and across time.

1 Introduction

Ground-to-aerial/satellite image matching, also known as cross-view image matching, has shown notable performance in large-scale geolocalization [8,15,16,25,34,37,40,48]. Usually, this global localization task is formulated as image retrieval. For each ground-level query image the system retrieves the most similar geo-tagged aerial/satellite patch in the database and uses the location of the center pixel in that patch as the location of the query. In practice, global localization can also be obtained by other means in outdoor robotics, such as temporal filtering or coarse GPS/GNSS [31,41,42], but can still have errors of

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-19842-7_6.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022
S. Avidan et al. (Eds.): ECCV 2022, LNCS 13699, pp. 90–106, 2022.
https://doi.org/10.1007/978-3-031-19842-7_6

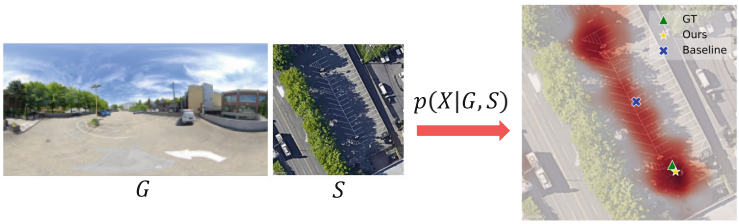

Fig. 1. Example of visual cross-view metric localization. Given a ground-level image G (left), and a satellite patch S (middle) with its local area, we aim to identify the location X within S where G was taken. Our method estimates a dense probability distribution over the satellite image. The resulting (log) probability heat map is overlayed in red on top of the satellite patch (right). Compared to the regression-based baseline that tends to roughly regress to the midpoint among multiple modes, our method captures the underlying multi-modal distribution. Our final predicted location, argmax$(p(X|G,S))$, is closer to the ground truth. (Color figure online)

tens of meters [4,41,42]. In this work, we therefore follow [31,41,42] by exploiting a coarse location estimate, and zoom into fine-grained *metric localization* within a known satellite image, i.e. to identify which image coordinates in the satellite patch correspond to the location of ground measurement. We adopt the common assumption [16,25,27,34,48] of known orientation, e.g. the center of a ground panorama points north, though we will seek to loosen this restriction in our experiments and roughly estimate the camera's heading too.

In vision, even though ground-to-ground metric localization is a well-studied task [1,6,13], so far in the cross-view setting, the only end-to-end approach that considers metric localization is the regression-based approach proposed in [48], which we will refer here to as *Cross-View Regression* (CVR) for simplicity. CVR tries to solve both the global coarse localization and local metric localization. As a result, its metric localization regressor is built on top of global image descriptors and might miss fine-grained scene information from the satellite image.

Rather than formulating visual cross-view metric localization as a regression task, we propose to produce a dense multi-modal distribution to capture localization ambiguities, and avoid regressing to the midpoint between multiple visually similar places, see Fig. 1. To capture more spatial information, we compute multiple local satellite image descriptors rather than a single global one, and train these in a locally discriminative manner. We note that dense uncertainty output for localization was shown to be successful with range-sensing modalities, like LiDAR and Radar, for localization within top-down maps [3,38,44]. However, these methods are not directly applicable to monocular vision, as they rely on highly accurate depth information which images lack.

Unlike existing literature [8,16,25,27,34,37,40,48], we address local metric localization as a standalone task in visual cross-view matching, and make the following contributions: (i) We propose to predict a dense multi-modal distribution for localization, which can represent localization ambiguity. For this, we propose a new Siamese-like network that exploits multiple local satellite descriptors and uses similarity matching in the fusion bottleneck. It combines the metric learning

paradigm from image retrieval with dense probabilistic output via a UNet-style decoder, found previously only in range-based cross-view localization. (ii) We show that the produced distribution correlates with localization quality, a desirable property for outlier detection, temporal filtering, and multi-sensor fusion. Besides, we also achieved significantly lower median localization error than the state-of-the-art. (iii) We show our proposed method is robust against small perturbations on the assumed orientation, and that the model's probabilistic output can even be used to classify a ground image orientation when it is unknown.

Our experiments use the recent large-scale VIGOR dataset for standalone cross-view metric localization to test generalization to new locations in both known and unknown areas. We also collect and stitch additional satellite data for data augmentation and metric localization on the Oxford RobotCar dataset, testing generalization to new measurements along the same route across time.[1]

2 Related Work

We here review the works most related to visual cross-view metric localization.

Cross-view Image Retrieval is a special case of image retrieval. For place recognition [17], a majority of works [5,35,36] construct a reference database using ground-level images, but it is infeasible to guarantee the coverage of the images everywhere. Alternatively, satellite images provide continuous coverage over the world and are publicly available. Given this advantage, a series of approaches [15,37,40] have been proposed to solve large-scale geolocalization using ground-to-satellite cross-view image retrieval. CVM-Net [8] adopts the powerful image descriptor NetVLAD [2] to summarize the view-point invariant information for the cross-view image retrieval. In [16], the authors encode the azimuth and altitude of the pixels in the ground-level query to guide the ground-to-satellite matching. To explicitly minimize the visual difference between satellite and ground domains, various improvements have been proposed. SAFA [25] proposes to use a polar transformation to warp the satellite patch towards the ground-level panorama and uses attention modules to extract the specific features that are visible from both views. In [21,34], a conditional GAN [10] is used to generate synthetic satellite images from the ground-level panorama or to synthesize the panoramic street view from the satellite image to direct the cross-view matching. Instead of constructing a visually similar input, CVFT [27] tries to transport the features from the ground domain towards the satellite domain inside an end-to-end network. Some works [26,37,47] jointly estimate the orientation of the ground query during retrieval without any metric localization. Recently, transformers [43,46] are also used in cross-view image retrieval.

Limitations in Cross-view Image Retrieval are also evident despite its increasing popularity for geolocalization. Recently, [48] points out that cross-view image retrieval methods assume that query ground images correspond to the center of satellite patches in the database, and this assumption is not valid

[1] Models and code, plus extended data are available at https://github.com/tudelft-iv/CrossViewMetricLocalization.

during test time. To break this assumption, [48] introduces a new cross-view matching benchmark VIGOR in which the ground images are not aligned with the center of satellite patches. Another limitation of retrieval is the trade-off between localization accuracy and computation or dataset density. To acquire meter-level localization accuracy, reference satellite patches often have a large overlap with each other, such as sampling the patch every 5m as done in [9,41].

Range Sensing Sensors-to-satellite Metric Localization received more attention than its visual counterpart. RSL-Net [31] localizes Radar scans on a known satellite image. This task is formulated as generating a top-down Radar scan conditioned on the satellite image using [10], and then comparing the online scan to synthetic scan for pose estimation. Later, this idea is extended to self-supervised learning [30]. In [29], the top-down representation of a LiDAR scan is compared to UNet [22] encoded satellite features for metric localization. The range information is crucial in representing the measurement in a top-down view.

LiDAR-to-BEV Map Metric Localization is another frontier that benefited from the range sensing. Dense pixel-to-pixel matchable LiDAR and bird's eye view (BEV) map embeddings can be learned by a deep network [3]. Localization becomes finding the position that has the maximum cross-correlation between two embeddings. Later work [38] shows that it is possible to localize the online LiDAR sweep on HD maps in a similar manner. Those works deliver a dense probabilistic output by formulating the localization task as a classification problem. This property is ideal in probabilistic robot localization [32], as it enables multi-sensor fusion and temporal filtering.

Visual Ground-to-satellite Metric Localization cannot directly reuse the same architecture used to localize LiDAR scans in a BEV map, since an RGB ground image does not provide reliable depth information. Hence pixel-level dense comparison, such as cross-correlation, cannot be leveraged. [45] predicts ground-view semantics from aerial imagery for orientation estimation, and shows only qualitatively that metric localization is possible by comparing the predicted semantics across viewpoints. To the best of our knowledge, CVR [48] is the only end-to-end approach in the vision domain that attempts metric localization on a satellite patch. Given a ground-level query, it first retrieves the matched satellite patch and then regresses the offset between the ground image and satellite patch center. However, its offset regression is based on global feature descriptors, which might cause the regression head to miss detailed scene layout information, and it limits the output to uni-modal estimates. Plus, CVR lacks dense uncertainty estimation to identify ambiguous locations, or a way to filter out unreliable results. A concurrent work [24] perform unimodal localization and orientation estimation by warping features across views and solving an iterative optimization.

3 Methodology

In our work, we assume that a rough prior localization estimate is available, e.g. through GPS/GNSS, odometry, or some other robot-localization techniques [31,41,42]. Given a ground-level image G and a top-down $L \times L$ satellite

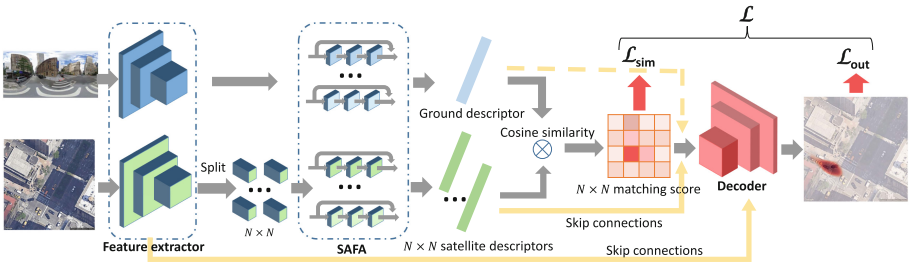

Fig. 2. An overview of the proposed cross-view metric localization architecture (trainable parts in bold). Dashed skip connection is optional, see ablation study. We overlay an exemplar output heat map on top of the input satellite image for intuition.

image S that represents the local area where G was taken, our metric localization objective is to estimate the 2D image coordinates $X \in [0, 1]^2$ within S that correspond to the ground location of the camera of G. Moreover, we aim for a dense probabilistic output to benefit a downstream sensor fusion task, similar to [3]. Note that in practice, G and S are often provided with their heading pre-aligned [16,48], such that the center vertical line of G points in the up direction of S.

Both the baseline CVR [48] and our proposed method adapt a common cross-view image retrieval architecture [25]. This basic backbone is a Siamese-like architecture without weight-sharing. Both the ground and satellite input branches consist of a VGG [28] feature extractor. E.g. for the satellite branch, these features form a $L' \times L' \times 512$ volume. On the feature volume 8 Spatial-Aware Feature Aggregation (SAFA) modules [25] are applied, each generating a 512-dimensional vector, which is all concatenated. Each branch thus yields a single global $1 \times 1 \times 4096$-dimensional descriptor. In an image retrieval task, this network would be trained through metric learning such that descriptors of matching (S, G) pairs are close together in the 4906-dimensional space.

Importantly, our proposed architecture and CVR make distinct choices on (1) the used descriptor representation for S, (2) how the descriptors are fused, (3) how the output head represents the localization result, and (4) consequently, the losses. We explain these choices for both methods in turn.

3.1 Baseline Cross-View Regression

The CVR method in [48] uses a single architecture for a two-step approach. First global localization is done through image retrieval by comparing descriptor G to descriptors of all known satellite patches. After retrieving satellite patch S, metric localization is performed using the already computed descriptors of both G and S. We employ CVR here for the metric localization task only, and therefore keep its proposed architecture, but will not train it for image retrieval. Focusing on metric localization only, our CVR baseline makes the following design choices:

Feature Descriptors: CVR follows the image-retrieval concept of encoding the satellite and ground image each into a single image-global 4096-dimensional descriptor. Both descriptors are fed as-is to the fusion step.

Fusion: CVR simply concatenates the two feature descriptors into a single 8192-dimensional vector.

Output Head: A multi-layer perceptron is used on the fused descriptors which outputs the relative 2D offset ΔX between G's true location within S and the center $X_S = (0.5, 0.5)$ of the satellite patch, s.t. $X = X_S + \Delta X$.

Loss: The standard L2 regression loss is used on the predicted offset and true offset.

We note that most of these choices follow from the need to use a single global descriptor for a whole satellite patch, as such descriptors are necessary for image retrieval. Our argument is however that if a localization prior is already available and global image retrieval is not necessary, this state-of-the-art architecture is sub-optimal for metric localization only compared to our proposed approach.

3.2 Proposed Method

Our proposed architecture starts with a mostly similar Siamese-like backbone. The method overview is shown in Fig. 2. It differs from CVR as follows:

Feature Descriptors: Instead of building one image-global descriptor to represent S, we increase the top-down spatial resolution by splitting the satellite $L' \times L' \times 512$ feature volume along spatial directions into $N \times N$ sub-volumes, where N is a hyper-parameter. Now the 8 SAFA [25] modules are applied to each $L'/N \times L'/N \times 512$ sub-volume in parallel, resulting in an $N \times N \times 4096$ descriptor $g(S)$ for the satellite branch, shown as the green vectors in Fig. 2. Let $g(S)^{ij}$ denote the i-th row j-th column of the satellite descriptor, $1 \leq i, j < N$. The ground image is still encoded as a single global 4096-dimensional descriptor $f(G)$, shown as the blue vector in Fig. 2.

Fusion: To help distinguish different satellite image sub-regions, we compute the cosine similarity between $f(G)$ and each $g(S)^{ij}$, and use this similarity as a feature itself at this fusion bottleneck. This similarity computation results in a $N \times N \times 1$ matching score map M, thus $M^{ij} = \text{sim}(f(G), g(S)^{ij})$. To complete our fusion step, the M is concatenated to the satellite descriptors $g(S)$ through a skip connection, shown as the upper yellow solid arrow in Fig. 2. Optionally, one could also concatenate $f(G)$ again into the fused descriptor (yellow dashed arrow), similar to CVR; we explore this in our experiments.

Output Head: Rather than treating metric localization as a regression task, we seek to generate a dense distribution over the image coordinates. Such output enables us to represent localization ambiguities and estimate the (un)certainty of our prediction. Towards this, we feed the fusion volume to a decoder which can progressively up-sample the $N \times N$ matching map to higher resolutions. Akin

to the UNet architecture [22], skip connections between satellite encoder and decoder are used to pass the fine-grained scene layout information to guide the decoding. Finally, a softmax activation function is applied on the last layer, and outputs a $L \times L \times 1$ heat map H, where each pixel $H^{u,v} = p(X \in c(u,v)|G,S)$ represents the probability of G being located within pixel area $c(u,v)$. This heat map is useful by itself, e.g. in a sensor fusion framework. For a single frame estimate, we simply output the center image coordinates $\overline{c}[\cdot]$ of the most probable pixel, i.e. $X = \overline{c}[\text{argmax}_{(u,v)} H^{u,v}]$.

Losses: A benefit of our framework is that we can add losses on both the final output and the fusion bottleneck. The full loss $\mathcal{L} = \mathcal{L}_{\text{out}} + \beta \times \mathcal{L}_{\text{sim}}$ is thus a weighted sum of the output loss, $\mathcal{L}_{\text{out}}$, and the bottleneck loss, $\mathcal{L}_{\text{sim}}$, where β is a hyper-parameter. We discuss each term next.

Since the output H is a discrete probability distribution that sums to one, we treat our task as a multi-class classification problem. $\mathcal{L}_{\text{out}}$ is simply a cross-entropy loss over the $L \times L$ output cells. The ground truth is one-hot encoded as a heat map with the same $L \times L$ resolution and label 1 at the true location and 0 elsewhere, In practice, we will apply Gaussian label smoothing to the one-hot encoding of the output head, and tune the smoothing σ as part of the hyperparameter optimization.

To guide the model to already learn locally discriminative satellite descriptors at the fusion bottleneck, we apply the infoNCE loss [20] from contrastive representation learning [11], which can be seen as a generalized version of triplet loss [23] used in image retrieval in the case of multiple negative samples are presented at the same time,

$$\mathcal{L}'(ij^+) = -\log \frac{\exp(\text{sim}(f(G), g(S)^{ij^+})/\tau)}{\sum_{i,j} \exp(\text{sim}(f(G), g(S)^{ij})/\tau)}. \tag{1}$$

Here τ is a hyper-parameter introduced by [20], and its role is similar to the margin between positive and negative samples in triplet loss, and (ij^+) is the cell index of the positive satellite descriptor w.r.t. the ground descriptor.

We reuse the smoothed one-hot encoding from the output loss to allow multiple soft positives if the true location is near a cell border. We max-pool the $L \times L$ target map to the $N \times N$ resolution and renormalize it to generate 'positiveness' weights w_{ij}^+ for each cell $1 \leq i, j \leq N$. Our bottleneck loss is simply a weighted version of Eq. (1), $\mathcal{L}_{\text{sim}} = \sum_{i,j} w_{ij}^+ \mathcal{L}'(ij)$.

4 Experiments

In this section, we first introduce the two datasets and evaluation metrics for our experiments. Then we motivate each of our design choices and provide a detailed ablation study. Finally, our model is compared to the baseline CVR approach [48] to show our advantage in metric localization in generalizing to new measurements in the same area, across areas, and across time.

4.1 Datasets

The first used dataset, **VIGOR** [48], contains geo-tagged ground-level panoramic images and satellite images collected in four cities in the US. Unlike previous cross-view image retrieval datasets [14,16,33,45], the satellite patches in VIGOR seamlessly cover the target area. Importantly, the ground-level panoramas are not located at the center of satellite patches. Each satellite patch corresponds to 72.96 × 72.96 m ground area with a ground resolution of 0.114 m. The orientation of the satellite patch and ground panorama are aligned in a way that the vertical line at the center of the panorama corresponds to the north direction in the satellite patch. Typically, each patch has ∼50% overlap with its neighboring patch in the North, South, East, and West direction. This means every ground image is covered by 4 satellite patches. If the ground image is at the center 1/4 area of a satellite patch, the patch is denoted as "positive", otherwise "semi-positive". In practice, "positive" samples simulate the case that the global localization prior is more accurate, e.g. error $< \sqrt{2} \times 18.24$ m in the case of VIGOR. Similarly, "positive + semi-positive" samples would be a result of a coarser localization prior, e.g. error $< \sqrt{2} \times 36.48$ m. During training, we include both "positives" and "semi-positives" samples. Our main evaluation will be based on positive samples, since it is representative for most real-world situations, e.g. localization prior from GNSS positioning in an open area or temporal filtering. For completeness, we also evaluate on "positive + semi-positive", to showcase how the methods behave with a less certain localization prior, e.g. GNSS positioning in an urban canyon. We adopt the "same-area" and "cross-area" splits from [48] to test the model's generalization in the same cities and across different cities. To find one set of hyper-parameters for both "same-area" and "cross-area", we create a subset of the shared training data from New York as a smaller "tuning" split with 11108/2777 training/validation samples.

The second dataset, **Oxford RobotCar** [18,19] contains multi-sensor measurements from multiple traversals over a consistent route through Oxford collected over a year. The original dataset does not contain satellite images. To enable cross-view metric localization, we stitch the satellite patches provided by [41,42] and our additionally collected ones to create a continuous satellite map that covers the target area. We follow the same data split as in [41] to test how our method generalizes to new ground images collected at different time. In total, there are 17067, 1698, and 5089 ground-level front-viewing images in the training, validation, and test set respectively. The test set contains 3 traversals collected at later times of day than the training recordings. Benefiting from a full continuous satellite map, we randomly and uniformly sample satellite patches around the ground image locations during training for data augmentation. Each patch is rotationally aligned with the view direction of the ground image and has a resolution of $800 pixels \times 800 pixels$, which corresponds to 73.92 × 73.92 m on the ground. A fixed set of satellite patches is used for validation and testing. Each patch has a 50% area overlap with the closest neighboring patches. We pair each ground image with the patch at the closest center location to allow for mutual information between the satellite and ground front-facing views.

Similarly, during training, we also control the sampled locations to make sure the ground image locates inside the central area of the sampled satellite patch.

4.2 Evaluation Metrics

To measure the localization error, we report the mean and median distance between the predicted location and ground truth location in meters over all samples. Note that the mean error can be biased by a few samples with large error, and including the median error provides a measurement more robust w.r.t. outliers. In practice, a localization method that operates on a single image frame can be extended to process a sequence of data using a Bayesian filter [9,41]. In such a setting, the estimated probability at the ground truth location plays an important role in accurately localizing over the whole path. Motivated by this, we include the probability at the ground truth pixel area as an additional metric. The baseline CVR method does not have any probability estimation on its output. Hence, we post-process the baseline output by assuming the regressed location is the mean of an isotropic Gaussian distribution, and we estimate the standard deviation of this Gauss on the validation set.

4.3 Hyper-parameters and Ablation Study

We first discuss our hyper-parameter choices, then investigate the main components in our proposed architecture. The weight β in our loss function is set to 10^4, and the τ in infoNCE loss is set to 0.1, as done in [20]. The loss is optimized by Adam optimizer [12] with a learning rate of 1×10^{-5}, and the VGG feature extractors are pre-trained on ImageNet [7]. Our main model variations and hyper-parameters are now compared on the VIGOR "tuning" split.

We initially set $N = 8$ for the matching at the bottleneck. The infoNCE loss at the bottleneck is key to improve the final model output. The model trained with it achieves a much better mean error, 14.30 m, than the model trained without it, 19.25 m. Label smoothing with $\sigma = 4$ pixels further reduces the mean error to 13.39 m (we tested $\sigma = 1, 2, 4, 8$). We use both in all future experiments.

Next, we study the influence of different resolutions $N \times N$ at the model bottleneck. When $N = 1, 2, 4, \mathbf{8}, 16$, the mean error is $19.62, 15.98, 15.23, \mathbf{13.39}, 15.04$ meters respectively (best in bold). With $N = 1$ no infoNCE loss is applied, and the decoder receives a single matching score concatenated with features from the satellite branch. Increasing N improves the spatial resolution at the model bottleneck. However, with larger N the decoder also operates on larger inputs but with fewer upsampling layers. We observe a balance at $N = 8$.

To further explore the role of metric learning at the bottleneck and the feature concatenation, we create four extra model variations in Table 1. Directly concatenating the ground with a single global satellite descriptor (see "1,S+G") is akin to CVR's fusion with a decoder head instead of regression, but this change alone does not perform well. The model does not work when the decoder

Table 1. Fusion bottleneck, error on tuning split: "S" stands for satellite descriptors $g(S)$, "G" for ground descriptors $f(G)$, "M" for cosine similarity feature. Best in bold

N, descriptors	1,S+G	8,M	8,S+G	8,S+M+G	8,S+M
Mean error (m)	18.62	24.37	18.35	13.83	**13.39**

Table 2. Localization error on VIGOR. Best in bold. "Center-only" denotes using satellite patch center as the prediction. The term "Positives" stands for evaluation on positive satellite patches. "Pos.+semi-pos." takes the mean over the results from the positive satellite patches and all semi-positive satellite patches

	Same-area				*Cross-area*			
	Positives		Pos.+semi-pos.		Positives		Pos.+semi-pos.	
Error(m)	Mean	Median	Mean	Median	Mean	Median	Mean	Median
Center-only	14.15	14.82	27.78	28.85	14.07	14.07	27.80	28.89
CVR [48]	10.55	9.31	16.64	13.82	**11.26**	10.02	18.66	16.73
Ours	**9.86**	**4.58**	**13.45**	**5.39**	13.06	**6.31**	**17.13**	**7.78**

operates on only a single channel map ("8,M") without any context from the satellite patch. Increasing the satellite resolution is also still insufficient with only ground descriptors ("8,S+G"), the descriptors must also be trained to be locally discriminative. Interestingly, we do not observe any benefit from also concatenating the ground descriptor ("8,S+M+G") to our default of satellite descriptors with matching scores ("8,S+M"). In all next experiments, we fuse only the satellite descriptors and the matching score.

We note that to forward-pass an input pair from VIGOR on a Tesla V100 GPU, CVR uses 0.020s, and our best-performing model 0.034s (i.e. ~ 30 FPS).

4.4 Generalization in the Same Area/Across Areas

We now compare our method to CVR on the VIGOR splits for generalizing to unseen ground images inside the same area, and across areas. When tested on the "same-area" correctly retrieved samples, CVR trained for only regression has a better mean (-1.5 m) and median (-1.1 m) localization error than CVR trained for both retrieval and regression, as expected. From now on, we will always train CVR model for regression only as the baseline.

Metric Error: The quantitative comparison against CVR on both VIGOR splits is summarized in Table 2. To highlight the value of conducting cross-view metric localization, we also include a "center-only" prediction which always outputs $X = (0.5, 0.5)$. Note that retrieval-only methods typically assume that the center of a satellite patch is representative of the true location.

When the ground image is compared to the positive-only satellite patches, our model reduces the median error by 51% over CVR when generalizing within the

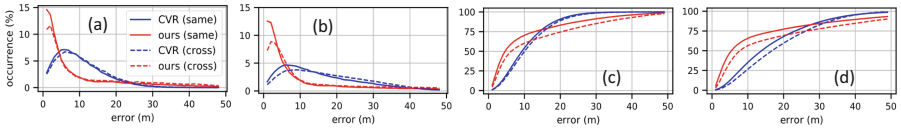

Fig. 3. Error distributions (plots a,b: regular, c,d: cumulative, a,c: positives, b,d: positives and semi-positives) on VIGOR, for same-area and cross-area experiments.

same area (4.58 m vs 9.31m), and by 37% when generalizing across areas (6.31 m vs 10.02 m). Generally, our model improves over the baselines, but across areas, our mean error is higher than that of CVR. The error (cumulative) distribution in Fig. 3 confirms that this is due to a few large-error outliers in our prediction. These outliers are a result of selecting a wrong mode, or of large uncertainty in our multi-modal output, whereas regression might pick an averaged location in the middle resulting in neither small nor very large errors. We will show below that our location's probability can be used to detect such potential large error cases. This would aid an external sensor fusion module, which can also directly integrate the distribution to reduce the uncertainty through other measurements.

When ground images could be located further from the center, as in the "positive+semi-positive" test cases, there is less matchable visual information between the two views. In this case, the performance of both CVR and our model somewhat degenerates, though our model suffers less than CVR. Our mean and median errors are lower than CVR's, both within the same area and across areas. Moreover, our method's advantage in the median error further increases. In the Sup. Mat. we furthermore investigate the effect of using a CVR-like regression layer and loss on top of our dense output but find it hurts median performance.

Qualitative Results: To intuitively understand where our advantage comes from, we provide qualitative examples of success and failure cases in Fig. 4. In the context of image-based cross-view metric localization, there can exist multiple visually similar locations on the satellite image given a ground image. In such cases, it is important for the model to have the capability to express the underlying localization uncertainty. In our model, the uncertainty is already present at the model bottleneck, see Fig. 4 top row. This distribution is upsampled by the decoder and aligned with the observed environmental features, such as roads and crossings, resulting in the dense multi-modal uncertainty map. We emphasize that no explicit semantic map information, e.g. on road layout, was used during training. Since the regression-based baseline method forces the output to be a single location, it risks 'averaging' multiple similar locations and provides a wrong final estimate without any uncertainty information.

We argue that in practice our outliers are still more acceptable than CVR's errors. When our model is uncertain about the exact location, our output heat map can be rather homogeneous. As shown in Fig. 4 example 3, given a ground image taken on the road, our model assigns high probability to roads in the center and on the left. In this situation, the distance between our predicted location and the ground truth can be large. Instead, CVR tends to output the

Fig. 4. Top: input ground images and matching score maps at the model bottleneck, bottom: input satellite image overlayed with outputs from CVR and our method. From left to right: 1: VIGOR, same-area, 2,3: VIGOR: cross-area, 4: Oxford RobotCar.

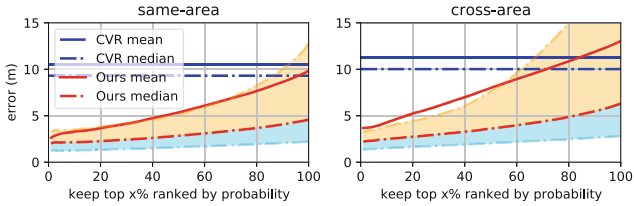

Fig. 5. Ranking the predictions using their probabilities on VIGOR "positives". Red lines show our error statistics over the top x%. Cyan/orange: error between median and 25%/75% quantile line. As x decreases, only the more probable predictions are kept. Blue lines: CVR cannot rank predictions this way. (Color figure online)

average between the visually similar areas, which can result in a location near the center but that is intuitively unreasonable, e.g. within some vegetation, even though it may have a smaller distance to the ground truth location.

Probability Evaluation: Apart from the metric localization error, we will compare how well each model can predict the probability at the ground truth location. For CVR we estimate this assuming a fixed Gaussian error distribution, see Sect. 4.2. Table 3 reports both mean and median probabilities at the ground truth pixel. Our multi-modal approach outperforms the fixed error distribution for CVR. Importantly, the probability at our predicted location (the maximum in H) is correlated to its localization error. If we apply a rejection threshold to only keep the top x%, predictions, we can reduce the expected error. See Fig. 5 with the statistics over the top-ranked estimates. These properties are beneficial when the single frame localization results are temporally filtered or fused with other sensors.

Orientation: Till now, we have relied on a known orientation during test time, e.g. estimated in the preceding retrieval step [26,37,47] or by the sensor stack [39]. We here study our model's robustness against orientation perturbations and ability to infer orientation when it is unknown *without retraining*. To

Table 3. Probabilities at the ground truth pixel on VIGOR. Best in bold. The magnitude of the probabilities is low due to the normalization over the 512×512 grid. "Uniform" shows for reference the prob. at GT for a homogeneous map, $1/(512 \times 512)$

Prob. at GT,	Same-area		Cross-area	
Positives	Mean	Median	Mean	Median
Uniform	3.81×10^{-6}	3.81×10^{-6}	3.81×10^{-6}	3.81×10^{-6}
CVR [48]	1.55×10^{-5}	1.70×10^{-5}	1.57×10^{-5}	1.72×10^{-5}
Ours	$\mathbf{2.93 \times 10^{-4}}$	$\mathbf{1.17 \times 10^{-4}}$	$\mathbf{1.54 \times 10^{-4}}$	$\mathbf{7.06 \times 10^{-5}}$

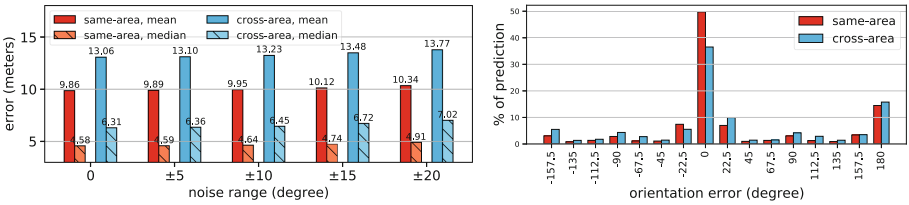

Fig. 6. Left: robustness of our model against small perturbations in orientation. Right: directly using our model to infer the unknown orientation.

test robustness, we uniformly sample angular noise from a range up to ±20° [27] to horizontally shift the ground panoramas (i.e. "rotate" the heading) at test time. As shown in Fig. 6 left, the predicted location of our model remains stable under such noise.

Still, the model's confidence is not invariant to orientation shift, as our prediction confidence can help classify a ground panorama's unknown orientation. We rotate the ground panorama by multiples of 22.5° up to 360°, apply our model to each rotated panorama with the satellite patch, and collect all 16 activation maps before the final softmax operation. The classification output is the orientation of the map with the highest activation. As shown in Fig. 6 right, for same and across areas, our model correctly classifies 50% and 37% samples into the true orientation class out of 16 classes. Most erroneous predictions have an error of 180°, corresponding to the opposite driving direction.

4.5 Generalization Across Time

Finally, we test how our method generalizes to new measurements collected at different times and days on the Oxford RobotCar dataset. For a comparison to cross-view image retrieval, we include "GeolocalRetrieval", which was previously also proposed on the Oxford RobotCar dataset [41]. While regular image retrieval is trained to be globally discriminative, this method learns descriptors that are only discriminative for nearby satellite patches within a 50 m radius, and thus assumes a localization prior during both training and testing, similar to our task. To increase its localization accuracy, we feed it a larger idealized dataset

Table 4. Localization error on Oxford RobotCar. Shown are the average ± standard deviation of 'mean' and 'median' errors over 3 test traversals. Best results in bold. ⋆: uses the same training and test ground images, but more overlapping satellite patches to obtain finer localization through image retrieval only

Error (meters)	Mean	Median
Center-only	12.09 ± 0.02	12.65 ± 0.01
GeolocalRetrieval⋆ [41]	6.01 ± 0.68	4.62 ± 0.49
CVR [48]	2.29 ± 0.31	1.72 ± 0.21
Ours	$\mathbf{1.77} \pm 0.25$	$\mathbf{1.24} \pm 0.10$

of satellite patches at more densely sampled locations (200+ patches in a 50 m radius) including even patches centered on the actual test locations. Therefore GeolocalRetrieval could obtain zero meter error if it correctly retrieves the exact satellite patch at each test image location.

Table 4 shows the localization error among all included methods. As expected, with the idealized satellite patches, GeolocalRetrieval delivers lower error than "center-only". However, metric localization methods (CVR and ours) show a clear advantage over GeolocalRetrieval. This highlights the benefit of conducting metric localization over simply densifying the dataset for retrieval. Moreover, using our model for metric localization reduces the mean error by 23% and the median error by 28% compared to using CVR. Qualitatively, we again observe the benefit of expressing multi-modal distribution over the CVR's regression even without the use of panoramic ground images, see Fig. 4 example 4. The probability evaluation also aligns with the findings on VIGOR. Our probability at the ground truth pixel are consistently higher than that under CVR with its estimated error distribution over three test traversals. Averaged over three test traversals, the mean/median probability at the ground truth pixel for CVR are $1.67 \times 10^{-4}/1.89 \times 10^{-4}$, and for ours are $1.54 \times 10^{-3}/1.38 \times 10^{-3}$.

We also test classification of the orientation on this non-panoramic dataset. Instead of shifting the ground image, we now rotate the satellite patch 16 times with 22.5°, starting at 0° where north points in the vertical up direction. The orientation of a ground image is inferred by selecting the peak probability as we did for VIGOR. On three test traversals, 72.3%, 70.7%, and 70.6% of the test samples are predicted with the correct orientation out of the 16 possible directions. More details on orientation classification and the localization results with unknown orientation can be found in the Supplementary Material.

To summarize, also for test images at new days our method shows all-round superiority, similar to generalization within the same and across areas.

5 Conclusion

In this work, we focused on visual cross-view metric localization on a known satellite image, a relatively unexplored task. In contrast to the state-of-the-art

regression-based baseline, our method provides a dense multi-modal spatial distribution. We studied the architectural design differences, and showed generalization to new measurements in the same area, across areas, and generalizing across time on two state-of-the-art datasets. Our method surpasses the state-of-the-art by 51%, 37%, and 28% respectively in the median localization error. In a few cases the multi-modal output yields higher distance errors, e.g. when an incorrect mode is deemed more probable. Still, our probabilities can be used to filter such large errors and have less risk of excluding the true location. We show that our method is robust against small orientation noise, and is capable to roughly classify the orientation from its prediction confidence. Future work will address temporal filtering and fine-grained orientation estimation.

Acknowledgements. This work is part of the research programme Efficient Deep Learning (EDL) with project number P16-25, which is (partly) financed by the Dutch Research Council (NWO).

References

1. Agarwal, P., Burgard, W., Spinello, L.: Metric localization using google street view. In: IEEE/RSJ IROS, pp. 3111–3118 (2015)
2. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: NetVLAD: CNN architecture for weakly supervised place recognition. In: Proceedings of IEEE/CVF CVPR, pp. 5297–5307 (2016)
3. Barsan, I.A., Wang, S., Pokrovsky, A., Urtasun, R.: Learning to localize using a lidar intensity map. In: CoRL (10 2018)
4. Ben-Moshe, B., Elkin, E., et al.: Improving accuracy of gnss devices in urban canyons. In: CCCG, pp. 511–515 (2011)
5. Chen, D.M., et al.: City-scale landmark identification on mobile devices. In: Proceedings of IEEE/CVF CVPR, pp. 737–744 (2011)
6. Clement, L., Gridseth, M., Tomasi, J., Kelly, J.: Learning matchable image transformations for long-term metric visual localization. IEEE Robot. Autom. Lett. **5**(2), 1492–1499 (2020). https://doi.org/10.1109/LRA.2020.2967659
7. Deng, J., Dong, W., Socher, R., et al.: Imagenet: A large-scale hierarchical image database. In: Proceedings of IEEE/CVF CVPR, pp. 248–255 (2009)
8. Hu, S., Feng, M., Nguyen, R.M., Hee Lee, G.: CVM-net: Cross-view matching network for image-based ground-to-aerial geo-localization. In: Proceedings of IEEE/CVF CVPR, pp. 7258–7267 (2018)
9. Hu, S., Lee, G.H.: Image-based geo-localization using satellite imagery. IJCV, pp. 1–15 (2019)
10. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of IEEE/CVF CVPR, pp. 1125–1134 (2017)
11. Khosla, P., et al.: Supervised contrastive learning. arXiv preprint arXiv:2004.11362 (2020)
12. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. ICLR (2014)
13. Lategahn, H., Stiller, C.: Vision-only localization. IEEE Trans. Intell. Transport. Syst. **15**(3), 1246–1257 (2014). https://doi.org/10.1109/TITS.2014.2298492

14. Lin, T.Y., Belongie, S., Hays, J.: Cross-view image geolocalization. In: Proceedings of IEEE/CVF CVPR, pp. 891–898 (2013)
15. Lin, T.Y., Cui, Y., Belongie, S., Hays, J.: Learning deep representations for ground-to-aerial geolocalization. In: Proceedings of IEEE/CVF CVPR, pp. 5007–5015 (2015)
16. Liu, L., Li, H.: Lending orientation to neural networks for cross-view geo-localization. In: Proceedings of IEEE/CVF CVPR, pp. 5624–5633 (2019)
17. Lowry, S., et al.: Visual place recognition: a survey. IEEE Trans. Robot. **32**(1), 1–19 (2015)
18. Maddern, W., Pascoe, G., Linegar, C., Newman, P.: 1 year, 1000 km: The oxford robotcar dataset. IJRR **36**(1), 3–15 (2017)
19. Maddern, W., Pascoe, G., et al.: Real-time kinematic ground truth for the oxford robotcar dataset. arXiv preprint: 2002.10152 (2020)
20. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
21. Regmi, K., Shah, M.: Bridging the domain gap for ground-to-aerial image matching. In: Proc. of IEEE/CVF ICCV, pp. 470–479 (2019)
22. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
23. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of IEEE/CVF CVPR, pp. 815–823 (2015)
24. Shi, Y., Li, H.: Beyond cross-view image retrieval: Highly accurate vehicle localization using satellite image. In: Proceedings of the IEEE/CVF CVPR (2022)
25. Shi, Y., Liu, L., Yu, X., Li, H.: Spatial-aware feature aggregation for image based cross-view geo-localization. In: NeurIPS, pp. 10090–10100 (2019)
26. Shi, Y., Yu, X., Campbell, D., Li, H.: Where am i looking at? joint location and orientation estimation by cross-view matching. In: Proceedings of IEEE/CVF CVPR, pp. 4064–4072 (2020)
27. Shi, Y., Yu, X., Liu, L., et al.: Optimal feature transport for cross-view image geo-localization. In: Proceedings of AAAI, pp. 11990–11997 (2020)
28. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
29. Tang, T.Y., De Martini, D., Newman, P.: Get to the point: Learning lidar place recognition and metric localisation using overhead imagery. Robotics: Science and Systems (2021)
30. Tang, T.Y., De Martini, D., Wu, S., Newman, P.: Self-supervised learning for using overhead imagery as maps in outdoor range sensor localization. IJRR **40**(12–14), 1488–1509 (2021)
31. Tang, T.Y., De Martini, D., Barnes, D., Newman, P.: Rsl-net: Localising in satellite images from a radar on the ground. IEEE Robot. Autom. Lett. **5**(2), 1087–1094 (2020)
32. Thrun, S., Burgard, W., Fox, D.: Probabilistic robotics. MIT press (2005)
33. Tian, Y., Chen, C., Shah, M.: Cross-view image matching for geo-localization in urban environments. In: Proceeidngs of IEEE/CVF CVPR, pp. 3608–3616 (2017)
34. Toker, A., Zhou, Q., Maximov, M., Leal-Taixe, L.: Coming down to earth: Satellite-to-street view synthesis for geo-localization. In: Proc. of IEEE/CVF CVPR. pp. 6488–6497 (June 2021)

35. Torii, A., Arandjelovic, R., Sivic, J., Okutomi, M., Pajdla, T.: 24/7 place recognition by view synthesis. In: Proceedings of IEEE/CVF CVPR, pp. 1808–1817 (2015)
36. Torii, A., Sivic, J., Okutomi, M., Pajdla, T.: Visual place recognition with repetitive structures. IEEE Trans. Pattern Anal. Mach. Intell. **37**(11), 2346–2359 (2015). https://doi.org/10.1109/TPAMI.2015.2409868
37. Vo, Nam N.., Hays, James: Localizing and orienting street views using overhead imagery. In: Leibe, Bastian, Matas, Jiri, Sebe, Nicu, Welling, Max (eds.) ECCV 2016. LNCS, vol. 9905, pp. 494–509. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_30
38. Wei, X., Bârsan, I.A., Wang, S., Martinez, J., Urtasun, R.: Learning to localize through compressed binary maps. In: Proceedings of IEEE/CVF CVPR, pp. 10316–10324 (2019)
39. Won, D., et al.: Performance improvement of inertial navigation system by using magnetometer with vehicle dynamic constraints. J. Sensors, 1–11 (2015)
40. Workman, S., Souvenir, R., Jacobs, N.: Wide-area image geolocalization with aerial reference imagery. In: Proceedings of IEEE/CVF ICCV, pp. 3961–3969 (2015)
41. Xia, Z., Booij, O., Manfredi, M., Kooij, J.F.P.: Cross-view matching for vehicle localization by learning geographically local representations. IEEE Robot. Autom. Lett. **6**(3), 5921–5928 (2021). https://doi.org/10.1109/LRA.2021.3088076
42. Xia, Z., Booij, O., Manfredi, M., Kooij, J.F.P.: Geographically local representation learning with a spatial prior for visual localization. In: Bartoli, A., Fusiello, A. (eds.) ECCV 2020. LNCS, vol. 12536, pp. 557–573. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-66096-3_38
43. Yang, H., Lu, X., Zhu, Y.: Cross-view geo-localization with layer-to-layer transformer. In: NeurIPS. pp. 29009–29020 (2021)
44. Yin, H., Chen, R., Wang, Y., Xiong, R.: Rall: end-to-end radar localization on lidar map using differentiable measurement model. IEEE Transactions on Intelligent Transportation Systems (2021)
45. Zhai, M., Bessinger, Z., Workman, S., Jacobs, N.: Predicting ground-level scene layout from aerial imagery. In: Proceedings of IEEE/CVF CVPR, pp. 867–875 (2017)
46. Zhu, S., Shah, M., Chen, C.: Transgeo: Transformer is all you need for cross-view image geo-localization. In: Proceedings of the IEEE/CVF CVPR, pp. 1162–1171 (2022)
47. Zhu, S., Yang, T., Chen, C.: Revisiting street-to-aerial view image geo-localization and orientation estimation. In: Proceedings of IEEE/CVF WACV, pp. 756–765 (2021)
48. Zhu, S., Yang, T., Chen, C.: Vigor: Cross-view image geo-localization beyond one-to-one retrieval. In: Proceedings of IEEE/CVF CVPR, pp. 3640–3649 (2021)