GLACE: Global Local Accelerated Coordinate Encoding

Fangjinhua Wang[1*] Xudong Jiang[1*] Silvano Galliani[2] Christoph Vogel[2] Marc Pollefeys[1,2]
[1]Department of Computer Science, ETH Zurich
[2]Microsoft Mixed Reality & AI Zurich Lab

Abstract

Scene coordinate regression (SCR) methods are a family of visual localization methods that directly regress 2D-3D matches for camera pose estimation. They are effective in small-scale scenes but face significant challenges in large-scale scenes that are further amplified in the absence of ground truth 3D point clouds for supervision. Here, the model can only rely on reprojection constraints and needs to implicitly triangulate the points. The challenges stem from a fundamental dilemma: The network has to be invariant to observations of the same landmark at different viewpoints and lighting conditions, etc.*, but at the same time discriminate unrelated but similar observations. The latter becomes more relevant and severe in larger scenes. In this work, we tackle this problem by introducing the concept of co-visibility to the network. We propose GLACE, which integrates pre-trained global and local encodings and enables SCR to scale to large scenes with only a single small-sized network. Specifically, we propose a novel feature diffusion technique that implicitly groups the reprojection constraints with co-visibility and avoids overfitting to trivial solutions. Additionally, our position decoder parameterizes the output positions for large-scale scenes more effectively. Without using 3D models or depth maps for supervision, our method achieves state-of-the-art results on large-scale scenes with a low-map-size model. On Cambridge landmarks, with a single model, we achieve 17% lower median position error than Poker, the ensemble variant of the state-of-the-art SCR method ACE. Code is available at:* https://github.com/cvg/glace.

1. Introduction

Visual localization describes the task of estimating the camera position and orientation for a query image in a known scene. This ability to localize in the environment is fundamental and important for applications like robotics, autonomous driving, and Augmented / Virtual Reality.

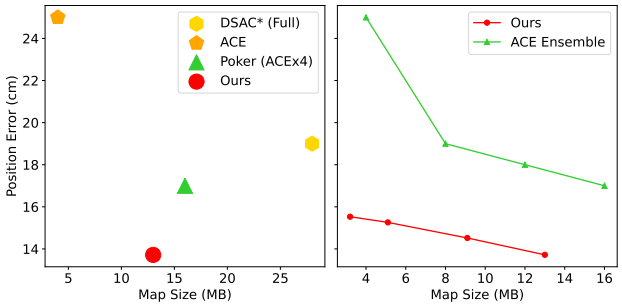

Figure 1. Left: Quantitative comparison of map size and position error with state-of-the-art SCR methods [9, 11] on Cambridge landmarks [24]. Our method outperforms DSAC [9], ACE [11] and Poker (4 ACE models) with a moderate model size. Right: Relationship between map size and position error. Note that our method with the smallest map size (3.2 MB) still performs better than Poker (4 ACE models, map size is 16.0 MB).

Currently, most state-of-the-art localization methods are structure-based [6, 9, 29, 30, 34, 43, 46], including feature matching based methods and most scene coordinate regression methods. Both techniques have in common to build maps from images with known poses. For localization they establish matches between 2D pixel positions in the query image and 3D points in the maps. Finally, embedded into RANSAC [3, 4], a Perspective-n-Point (PnP) solver [13, 21] is used to predict the camera pose from the 2D-3D correspondences. Both methodologies differ in the representation of the map and the estimation of correspondences.

Given a database of images, methods based on feature matching [29, 30, 34, 39, 43, 44] typically represent the 3D scene by reconstructing the 3D geometry, *e.g.* point cloud, using structure-from-motion (SfM) [38]. At test time, they establish 2D-3D matches between pixels in a query image and 3D points in the 3D model using descriptor matching. However, these methods need to store point-wise visual descriptors for the whole point cloud, which may cause storage issues when the scenes scale up.

In contrast, scene coordinate regression (SCR) methods [5, 9, 11, 16, 20] implicitly encode the map information inside a deep neural network. Instead of computing 2D-3D matches via explicit descriptor matching, these meth-

*Equal contribution.

ods directly regress the matches. Though achieving superior performance in small scenes [41], it is difficult to scale these methods to large-scale scenes due to the limited capacity of a single network [9]. A common solution is to train multiple networks on sub-regions of the scene [6]. But this certainly increases the model size, training time, and query time. Recent works [5, 7, 11] avoid the need for depth maps or a complete 3D model for training. In addition, ACE [11] proposes a method to train a 4MB-sized network in 5 minutes, while achieving state-of-the-art performance for smaller scenes [41]. Although it has impressive efficiency, ACE [11] still possesses the same problem of scaling to larger problem sizes and requires the use of an ensemble of networks for large-scale scenes [24], which lessens efficiency and practicality.

In this work, we propose GLACE, a novel method that enables the scene regression methodology to work on large-scale scenes with only a single network. Our method achieves state-of-the-art results on several large-scale datasets [6, 24] while using only a single model of small size and without using 3D models for supervision. Our contribution can be summarized as follows:

i) To our knowledge, our method is the first attempt of an SCR method to achieve state-of-the-art performance on large-scale scenes without using an ensemble of networks or 3D model supervision.

ii) We propose a novel feature diffusion technique for the pre-trained global encodings that implicitly groups the reprojection constraints with co-visibility, which avoids overfitting to trivial solutions.

iii) We propose a positional decoder that parameterizes the output positions for large-scale scenes more effectively than previous work.

2. Related Work

Pose Regression. Pose regression approaches [12, 23, 24, 26, 40, 48, 54, 56] encode the scene into a neural network and are trained end-to-end. At test time they regress an absolute or relative pose from a query image. Without geometric constraints, absolute pose regression methods [23, 24, 26, 51] usually do not generalize well to novel viewpoints or appearances. Besides, these methods do not scale well when limiting network capacity [43]. Operating differently, relative pose regression methods [2, 19, 56] regress a camera pose relative to one or more database images. While being scene-agnostic, they are often limited in accuracy.

Feature Matching Based Localization. Localization methods based on feature matching (FM) [29, 30, 33, 34, 39, 42–44] are often still considered state-of-the-art for visual localization. Those methods establish 2D-3D correspondences between pixels in a query image and 3D points in a scene model using descriptor matching. To scale to large scenes and handle challenging problems, such as day-night illumination change and seasonal change, these methods first perform a form of coarse localization. For instance, using image retrieval [1, 47], to first identify a small set of potentially relevant database images and only then perform descriptor matching with the 3D points visible in these images. However, these methods need to store all the descriptor vectors of the 3D model to perform matching, which may cause storage issues for large maps. Recently, several works [27, 57] try to avoid storing descriptors explicitly and instead propose to match directly against the geometry, *e.g.*, given as point cloud or mesh.

Scene Coordinate Regression. Given a query image, this family of localization methods regresses for a 2D pixel the corresponding 3D coordinates in the scene [41]. Usually, these methods implicitly store the information about the scene within the weights of a machine learning model. To regress 2D-3D matches, SCR methods are mainly based on random forests [8, 15, 17, 41, 49] or convolutional neural networks [5–7, 9, 11, 20, 25]. Recently, ACE [11] only uses posed RGB images for mapping. The training is performed only from the images using a loss based on the image reprojection error while completely avoiding the explicit reconstruction of a 3D model. It achieves state-of-the-art performance on several small-scale scenes [41, 50], and demonstrates impressive efficiency in training time and map size. However, a single model based on SCR is usually limited to only working on scenes of small-scale [9]. Larger scenes require techniques like an ensemble of SCR networks [6, 11] to scale, which demands additional maintenance, training time, and memory. In contrast, our method scales SCR methods to large-scale scenes without requiring an ensemble of networks or 3D model supervision.

3. Method

In this section, we first introduce the basic concepts for scene coordinate regression with ACE [11]. We follow by discussing how the system performs implicit triangulation when training without ground truth scene coordinate supervision and describe challenges in large-scale scenes for SCR methods. We conclude by introducing co-visibility to SCR in the form of global encodings and explain how we effectively enable the network to utilize this information. Finally, we discuss our novel position decoding technique that removes a bias in the SCR toward producing solutions near the center of training camera positions.

3.1. Scene Coordinate Regression

Visual Localization. We consider visual localization from RGB images. For training, we require a set of images with corresponding ground truth poses$\{(I_{train}, h_{train})\}$, where

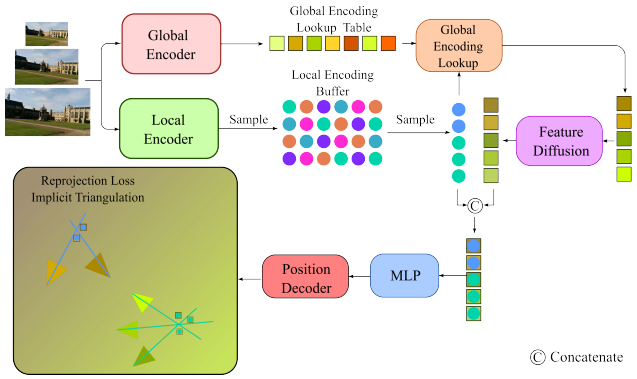

Figure 2. Pipleine of GLACE. Besides the buffer of ACE [11] local encodings, we extract global features of training images with image retrieval model [58]. During training, we sample a batch of local encodings, look up their global encoding according to their image index and perform feature diffusion by adding Gaussian noise. The global and local encodings are concatenated as input to an MLP head. The output of the MLP is further processed by a position decoder to yield the final coordinate predictions. The global encoding with feature diffusion facilitates the grouping of reprojection constraints, enabling effective implicit triangulation in large-scale scenes. Best viewed when zoomed in.

h_{train} denotes the rigid transformation from world coordinates to camera coordinates. During testing, our system estimates the camera pose h_{test} for a query image I_{test}. To that end, we follow the SCR methodology. Specifically, we mainly consider the setting with a pretrained local feature extractor and without ground truth scene coordinate supervision, established by ACE [11]. We first briefly review the SCR pipeline.

SCR Pipeline. SCR methods belong to the structure-based methods, which first predict 2D-3D correspondences and then solve for the pose with PnP and RANSAC. Traditional structure-based methods usually explicitly store a triangulated point cloud with corresponding features and match them with query image features to obtain 2D-3D correspondences. Instead, SCR methods implicitly learn the 2D-3D correspondence, usually in a convolutional neural network, which outputs the corresponding 3D coordinate for each image patch:

$$y_i = f(p_i), \tag{1}$$

where p_i is the image patch centered at pixel x_i and y_i is the corresponding 3D coordinate, the function f is given by the neural network. Previous works [6, 9, 25, 52] supervise the output y_i by providing ground truth 3D scene coordinates, *e.g.*, from a depth sensor or an SfM point cloud.

SCR with Reprojection Loss. Some recent works [5, 11] enable training without ground truth scene coordinates by employing a reprojection loss:

$$e_\pi(x_i, y_i, h) = ||x_i - \pi(\mathbf{K} \cdot h \cdot y_i)||_1, \tag{2}$$

where h is the ground truth pose of the image $\mathbf{K}$ is the camera intrinsic matrix and π performs the mapping from homogeneous to pixel space. The reprojection loss is usually combined with a robust loss function to reduce the influence of outliers. We use the dynamic tanh loss introduced in ACE [11]:

$$l_\pi(x_i, y_i, h) = \begin{cases} \tau(t) \tanh(\frac{e_\pi(x_i, y_i, h)}{\tau(t)}), & \text{if } y_i \in V \\ ||y_i - \bar{y}_i||_1, & \text{otherwise} \end{cases} \tag{3}$$

where V is the set of valid predictions, defined as points that are between 0.1m to 1000m in front of the camera and have a reprojection error $e_\pi(x_i, y_i, h)$ less than 1000px. $\bar{y}_i$ is the pseudo ground truth scene coordinate defined by the inverse projection of the pixel with the ground truth pose and a fixed target depth at 10m. During training the threshold $\tau(t)$ is adjusted dynamically based on the relative training time t:

$$\tau(t) = \sqrt{1 - t^2}\tau_{max} + \tau_{min}. \tag{4}$$

Reprojection Loss as Implicit Triangulation. In standard reconstruction, 2D-3D correspondences are explicitly established through matching. Observations of the same 3D point are grouped into a track, and the 3D point is triangulated by minimizing their reprojection error. In contrast, in SCR methods such as ACE [11] and ours, there is no explicit grouping of 2D observations for the same 3D point. Instead, each 2D observation *independently* regresses to a 3D point. Though initially seems like an underdetermined problem, these methods demonstrate practical efficacy, which we attribute to an *implicit triangulation* process. This process is driven by the inherent prior of neural networks to deliver smooth functions [28, 45], where similar inputs tend to produce similar outputs and undergo similar supervision. Thus, the reprojection loss for similar inputs is collectively minimized, leading to the triangulation of their corresponding output points. This insight explains the practical success of such methods, but also underlines the problem of applying SCR on large maps, which possess unrelated, yet, visually similar image observations and provides the motivation for our feature diffusion techniques.

3.2. Global Local Encoding

Challenges in Large-scale Scenes. SCR methods possess state-of-the-art accuracy in small indoor scenes. However, they struggle in larger environments, especially when no ground truth scene coordinate supervision is available, and the network needs to perform implicit triangulation of the coordinates from scratch. Consider the trade-off between invariance and discriminative power. Specifically, the dilemma of the receptive field. A model with a smaller receptive field satisfies the invariance assumption better and

can generalize to new observations, but suffers from ambiguity when there are different locations with similar local appearances, which intuitively occurs more frequently in large-scale scenes. On the other hand, a model with a larger receptive field may be able to disambiguate similar patches in different locations, but this breaks the invariance assumption: the network will also distinguish observations of the same scene coordinate. This can lead to overfitting to trivial solutions, *e.g.*, producing an arbitrary point along the ray, instead of triangulating the point from different observations. This also leads to poor generalization, since novel views of a point observed in training cannot be associated with it anymore.

Global Encoding. In order to solve the dilemma, we propose to carefully introduce global information, only including what is necessary. First, we analyze what exactly is needed from global information. Without global information, ambiguous patches that may belong to different scene points will together, via the reprojection loss, affect the triangulation of the same point. Also, the robustness in the loss function Eq. 3 can only mitigate, but not solve such problems. Therefore, we need global information to effectively group the reprojection constraints. Specifically, we only want to triangulate points in two images if and only if they are looking at the same thing, *i.e.* the views share sufficient co-visible structure. To effectively measure co-visibility, we utilize a global feature from an image retrieval model R^2Former [58], pretrained on MSLS dataset [53] and supervised by triplet margin loss with margin m:

$$L_{retrieval} = \max(||E_q - E_p||^2 - ||E_q - E_n||^2 + m, 0), \quad (5)$$

where E_q, E_p, E_n are global features of query, positive, and negative samples. The global features are 256-dimensional vectors normalized to the unit sphere. Here, we further analyze the relationship between feature distance and co-visibility using the SfM reconstruction of a large scene. A generative modeling analysis in Fig. 3 depicts the distribution of the angular feature distance(°) $d = \frac{180}{\pi} \arccos(u \cdot v)$ conditioned on the number of co-visible points. It strongly reminds us of a mixture of Gaussians, where the distribution of co-visible pairs possesses a lower mean. The discriminative model in Fig. 4 shows that the conditional probability of co-visibility c conditioned on feature distance d resembles $P(c|d) \sim \text{Bernoulli}(p)$, where the parameter p equals a sigmoid-like function of the feature distance. p is high before a 'threshold', then starts to decrease quickly to a low level afterward, which implies that it is possible to discriminate co-visibility based on the feature distance.

Naive Concatenation. With the co-visibility information contained in the global encoding, we still need to effectively integrate global and local features. First, consider the naive concatenation. In our discussion above, we assume that patches with similar input encoding will triangulate the same point together. When we concatenate local and global encoding together, inputs will triangulate the same point when both the local and global encoding are similar. However, as shown in Fig. 3, the feature distance between co-visible image pairs, although generally smaller than non-covisible pairs, may still be quite large. Intuitively, views of the same point with some angle between them will only partially overlap and thus possess global descriptors that do not match as well as local descriptors, and the concatenated descriptors of matching patches will not have a small distance as before. Those images that have almost the same global encoding, possess a very similar pose, with a small baseline, and contribute only little to the triangulation. Hence, if we simply concatenate global local encodings together, only a few images with small baselines are grouped together, which leads to large triangulation errors. Furthermore, the network might struggle to associate unseen views (w.r.t. to spatial coverage) during testing and generalize badly.

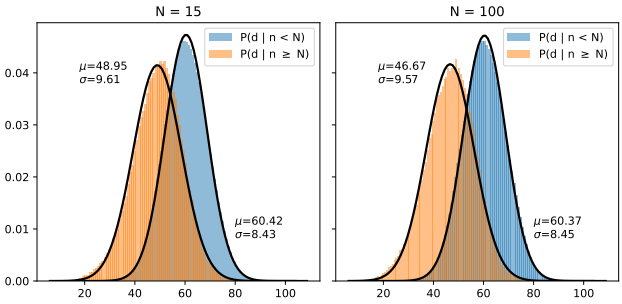

Figure 3. Distribution of angular feature distance(°), conditioned on co-visibility. Two images are considered co-visible, if the number of co-visible points n at least reaches a threshold N. The x-axis depicts the angular distance d in degrees (*left*: N=15, *right*: N=100).

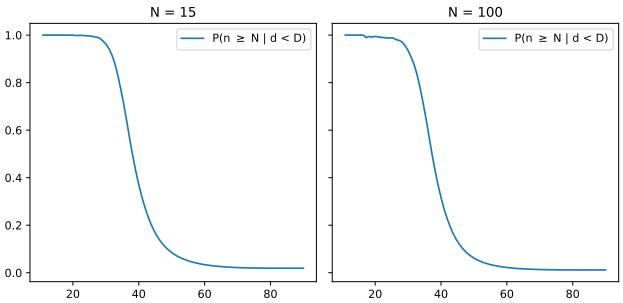

Figure 4. Distribution of co-visibility conditioned on the angular feature distance(°). Two images are considered co-visible, if the number of co-visible points n at least reaches a threshold N. The x-axis depicts the angular threshold D (*left*: N=15, *right*: N=100).

Explicit Clustering. A simple idea to solve this problem is

to explicitly cluster the global features, associate each feature with its cluster center, and use this as global encoding. This forces a grouping into 'hard' clusters of features with the same global encoding. However, this hard clustering approach requires to decide on an appropriate number of clusters. The number has to be large enough to ensure each cluster has a sufficient number of observations per point for triangulation and small enough to avoid ambiguous local encodings within a cluster, as shown in Tab. 6.

Implicit Grouping with Feature Diffusion. We propose a novel feature diffusion technique to perform the grouping implicitly. The idea is simple: instead of using a single fixed global feature for each image, we add some noise to make it a distribution. For the simplicity of sampling, we add Gaussian noise with a standard deviation of $\sigma = m$, where m is the margin for the image retrieval loss in Eq. 5. After adding the noise, the encoding is mapped back to the unit sphere. This method can be viewed as a form of feature metric data augmentation that imposes a stronger smoothness prior on global encoding, which prevents the neural network from easily discriminating co-visible pairs, thereby promoting implicit triangulation. Distinct from traditional image metric augmentations that typically involve alterations in the input image space, such as color jittering. Our approach operates directly within the feature space, where distances more accurately reflect covisibility relationships. The choice of hyperparameters, grounded in the metric space properties of the pretrained encoder, eliminates the need for scene-specific tuning, thereby ensuring robust performance across different scenes.

3.3. Position Decoding

Research [37] shows that the final layer has an important effect on the prior of CNNs that regress spatial positions, if the direct output of the last linear layer is a linear combination of bases in its weight. Therefore, it is important to effectively parameterize the final position by the network output, especially when there is no ground truth scene coordinate supervision, and we rely on the prior of the model to perform implicit triangulation. The network output of ACE [11] $(\dot{d}, \hat{w})$ defines an offset in homogeneous coordinates from the center of training camera positions c:

$$\hat{y} = \frac{\dot{d}}{w} + c. \tag{6}$$

$$w = \min\left(\frac{1}{S_{\min}}, \beta^{-1}\log(1+\exp(\beta\hat{w})) + \frac{1}{S_{\max}}\right). \tag{7}$$

$S_{\min}, S_{\max}$ are hyperparameters that define minimum and maximum scale and $\beta = \frac{\log 2}{1 - S_{\max}^{-1}}$ is the parameter for the softplus. It can better parameterize points at different scales,

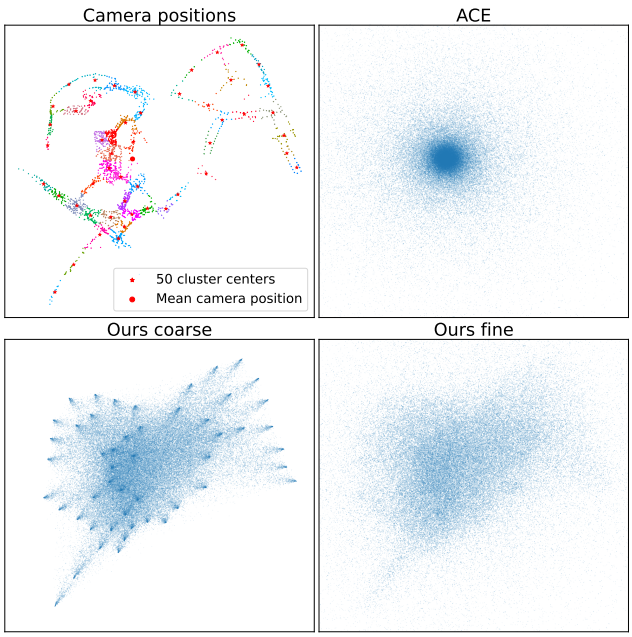

Figure 5. Comparison between decoder output of random Gaussian input samples. We use 50 cluster centers in this example of the Aachen dataset, shown in the top left (cluster assignments are color-coded, and cluster centers occur as *red star*).

but still suffers from *an unimodal prior*, preferring localization near the center c (Fig. 5, top right). Here, we propose an effective position decoder that predicts a convex combination of cluster center positions to replace the fixed center c in Eq. 6. We use K-Means to distribute the training camera positions into k clusters with centers $\{c_i\}$. The final linear layer of our MLP outputs k logits $\{s_i\}$, one for each cluster center and one homogeneous coordinate with parameters $\dot{d}, \hat{w}$ to define an offset. The final output is calculated similarly to Eq. 6. We only replace the center of training camera positions with the convex combination (using the softmax of logits) of cluster centers:

$$\hat{y} = \frac{\dot{d}}{\hat{w}} + \sum_{i=1}^{k} \frac{e^{s_i}}{\sum_j e^{s_j}} c_i. \tag{8}$$

We demonstrate the idea of our model and compare it to the encoding of [11] in Fig. 5. We sample from a Gaussian distribution as input and compare the decoded output for different decoders. Because of the unimodal prior of the ACE decoder, most of the samples are concentrated at the center. As a convex combination of clusters centers our model is inherently multimodal, but the samples are still concentrated at the modes. After adding the offset, the samples are distributed more evenly (Fig. 5, bottom right). Although the output of an MLP may not be a simple Gaussian distribution, this still can show that our decoder can better parameterize the output. We also designed a simple toy ex-

		Mapping w/ Mesh/Depth	Map Size	7 Scenes		12 Scenes	
				SfM poses	D-SLAM poses	SfM poses	D-SLAM poses
FM	AS (SIFT) [35]	No	∼200MB	98.5%	68.7%	99.8%	99.6%
	D.VLAD+R2D2 [22]	No	∼1GB	95.7%	77.6%	99.9%	99.7%
	hLoc (SP+SG) [29, 30]	No	∼2GB	95.7%	76.8%	100%	99.8%
	pixLoc [31]	No	∼1GB	N/A	75.7%	N/A	N/A
SCR (w/ Depth)	DSAC* (Full) [7]	Yes	28MB	98.2%	84.0%	99.8%	99.2%
	DSAC* (Tiny) [7]	Yes	4MB	85.6%	70.0%	84.4%	83.1%
	SANet [55]	Yes	∼550MB	N/A	68.2%	N/A	N/A
	SRC [20]	Yes	40MB	81.1%	55.2%	N/A	N/A
SCR	DSAC* (Full) [7]	No	28MB	96.0%	81.1%	99.6%	98.8%
	DSAC* (Tiny) [7]	No	4MB	84.3%	69.1%	81.9%	81.6%
	ACE [11]	No	4MB	**97.1%**	80.8%	99.9%	**99.6%**
	GLACE (Ours)	No	9MB	95.6%	**81.4%**	**100%**	**99.6%**

Table 1. **Quantitative results for single scene relocalization.** We report the percentage of frames below a 5cm,5° pose error. For the "SCR" group, best results in **bold**, second best results underlined. We list the map size and whether depth (rendered or measured) is needed for mapping.

Method	w / Depth	Size	i12	i19
ESAC [6]	Yes	336/532MB	97.1%	88.1%
ACE [11]	No	4MB	10.3%	5.9%
ACE [11] × 4	No	16MB	77.4%	36.5%
ACE [11] × 19	No	78MB	99.3%	90.9%
GLACE (Ours)	No	9MB	99.1%	87.0%

Table 2. **Integrated rooms dataset evaluation** with D-SLAM poses. We report the percentage of frames below a 5cm,5° pose error. ESAC uses 12/19 ensembles and has map size 336/532MB respectively.

Method	Size	i12	i19
ACE [11]	4MB	9.0%	17.0%
ACE [11] × 4	16MB	76.9%	42.0%
ACE [11] × 19	78MB	99.9%	97.8%
GLACE (Ours)	9MB	99.5%	93.4%

Table 3. **Integrated rooms dataset evaluation** with SfM poses. We report the percentage of frames below a 5cm, 5° pose error.

periment in supplementary material using a simplified 2D task that predicts the coordinates of the center pixel of a 2D image patch. The results show that even with strong supervision, the original decoder cannot regress the coordinate well when the scale is large. In contrast, with the help of our positional decoder, the performance is improved significantly. Please refer to the supplement for details.

4. Experiment

4.1. Datasets

7 Scenes [41] and 12 Scenes [50] are two standard datasets for room-scale indoor RGB-D localization. They contain 7 and 12 scenes respectively, each with a set of RGB-D sequences. There are two sets of ground truth poses for each scene, one from SfM and one from depth-based SLAM. Since they both have some bias [10], we report results on both of them following prior work [11]. To evaluate localization in large-scale indoor scenes, previous works [6, 52] have proposed to integrate multiple rooms from 7 Scenes and 12 Scenes into a single scene, denoted by i7, i12, and i19. We strictly follow [6], placing the scenes inside a 2D grid with a cell size of $5m$.

Cambridge Landmarks [24] is a large-scale outdoor dataset, with RGB sequences of landmarks in Cambridge. It includes ground truth poses and a sparse 3D reconstruction generated via SfM. The dataset is notable for its large-scale and outdoor setting, providing a different set of challenges compared to small-scale indoor datasets.

The Aachen Day-Night dataset [32, 36] is a city-scale dataset, which is particularly challenging for SCR methods due to its large scale and sparsity. It contains only limited images of Aachen city and ground truth poses provided via SfM. Here, we only consider Aachen Day, because there is no night-time training data.

4.2. Implementation

Architecture. We implement our method in PyTorch based on the official implementation of ACE [11]. The MLP architecture is the same as ACE[11], except that the network width is adjusted to match the input dimension of concatenated encoding. In addition, we use more residual blocks and increase the hidden size of the residual block for large outdoor scenes such as Cambridge and Aachen to increase model capacity, while still maintaining a comparable map size as baseline methods. We also tried concatenating the Superpoint [18] descriptor to the original ACE local encoder for the Aachen dataset to provide a more discrimi-

		Mapping w/ Mesh/Depth	Map Size	Cambridge Landmarks					Average (cm / °)
				Court	King's	Hospital	Shop	St. Mary's	
FM	AS (SIFT) [35]	No	∼200MB	24/0.1	13/0.2	20/0.4	4/0.2	8/0.3	14/0.2
	hLoc (SP+SG) [29, 30]	No	∼800MB	16/0.1	12/0.2	15/0.3	4/0.2	7/0.2	11/0.2
	pixLoc [31]	No	∼600MB	30/0.1	14/0.2	16/0.3	5/0.2	10/0.3	15/0.2
	GoMatch [57]	No	∼12MB	N/A	25/0.6	283/8.1	48/4.8	335/9.9	N/A
	HybridSC [14]	No	∼1MB	N/A	81/0.6	75/1.0	19/0.5	50/0.5	N/A
APR	PoseNet17 [23]	No	50MB	683/3.5	88/1.0	320/3.3	88/3.8	157/3.3	267/3.0
	MS-Transformer [40]	No	∼18MB	N/A	83/1.5	181/2.4	86/3.1	162/4.0	N/A
SCR w/ Depth	DSAC* (Full) [7]	Yes	28MB	49/0.3	15/0.3	21/0.4	5/0.3	13/0.4	21/0.3
	SANet [55]	Yes	∼260MB	328/2.0	32/0.5	32/0.5	10/0.5	16/0.6	84/0.8
	SRC [20]	Yes	40MB	81/0.5	39/0.7	38/0.5	19/1.0	31/1.0	42/0.7
SCR	DSAC* (Full) [7]	No	28MB	34/0.2	**18/0.3**	21/0.4	5/0.3	15/0.6	19/0.4
	DSAC* (Tiny) [7]	No	4MB	98/0.5	27/0.4	33/0.6	11/0.5	56/1.8	45/0.8
	ACE [11]	No	4MB	43/0.2	28/0.4	31/0.6	5/0.3	18/0.6	25/0.4
	Poker (ACE [11] × 4)	No	16MB	28/0.1	**18/0.3**	25/0.5	5/0.3	**9/0.3**	17/0.3
	GLACE (ours)	No	13MB	**19/0.1**	19/0.3	**17/0.4**	**4/0.2**	**9/0.3**	**14/0.3**

Table 4. **Cambridge Landmarks [24] Results.** We report median rotation and position errors. Best results in **bold** for the "SCR" group, second best results underlined.

native local descriptor.

Training. Most of the training parameter choices are the same as ACE, but we use larger buffer sizes for larger scenes, because there is more training data to be cached. In addition, we also use a larger batch size. As shown in Sec. 3.1, the reprojection supervision acts as an implicit triangulation. Therefore, it is desirable to have multiple observations of the same point in one batch to get stable and accurate supervision. In order to cache these larger buffers, we use distributed training with multiple GPUs. Specifically, we use a batch size of 160K and a training buffer size of 64M for the Cambridge dataset, a batch size of 320K and a training buffer size of 128M for Aachen and i19. For the Superpoint [18] version on Aachen, we also perform importance sampling according to its corner detection likelihood in order to select more salient structures. We train 30k iterations for Cambridge and 100k iterations for Aachen and i19.

4.3. Evaluation Results

7 Scenes and 12 Scenes. As indicated in Tab. 1, our approach retains the benefits of accuracy and compact map size observed in SCR methods when applied to small room-scale scenes.

Integrated Rooms. As shown in Tab. 2 and Tab. 3, previous SCR methods need a much larger map size, or demand an ensemble of networks in order to achieve satisfactory performance on large indoor scenes. Our method achieves comparable performance by a single model with a much smaller total map size. During test time, we only need to query a single model instead of all the ensemble models, which also makes our method more efficient and practical.

Cambridge Landmarks. This real-world outdoor dataset can fully demonstrate the advantages of our method. As shown in Tab. 4, our method significantly outperforms state-of-the-art SCR methods [9, 11] and closes the gap with FM methods [29, 31]. Particularly, the largest scene in this dataset, GreatCourt, is very challenging for SCR methods, but our method can still achieve comparable performance to FM methods with a small model size.

Aachen Day. We also evaluate our method on the Aachen dataset. The challenges of this dataset are not only the scale but also the sparsity. There are only about 4K discrete images for a city-scale scene, while the other datasets consist of several sequences with thousands of images for a small scene. Previous methods [6] usually rely on the ground truth scene coordinate supervision, however, we can still achieve comparable results without ground truth scene coordinate supervision and a much smaller map size. Other methods [11] that also feature small map sizes and no scene coordinate supervision will fail with a similar map size as ours. They cannot achieve a similar performance even with an ensemble of 50 models. In addition, we also tried concatenating SuperPoint [18] features to the original ACE [11] local features to increase the discriminative power and achieved better performance with smaller map size.

4.4. Ablation Study

Feature Diffusion. Tab. 6 compares different kinds of global encoding input on Cambridge Landmarks [24] and shows the effectiveness of our feature diffusion technique. When we directly concatenate the global encoding to the local encoding, the performance suffers from overfitting, especially apparent for simple scenes like StMarysChurch. If

Method	w / Depth	Size	0.25m,2°	0.5m,5°	5m,10°
ESAC × 50 [6]	Yes	1400MB	42.6%	59.6%	75.5%
ACE [11] × 4	No	16MB	0.0%	0.5%	3.8%
ACE [11] × 50	No	205MB	6.9%	17.2%	50.0%
GLACE (Ours)	No	27MB	8.6%	20.8%	64.0%
GLACE (Ours, SuperPoint [18])	No	23MB	9.8%	23.9%	65.9%

Table 5. **Aachen Day evaluation.** We compare the accuracy on Aachen Day dataset [32, 36].

Scene	ACE [11]	Poker (ACE × 4)	GLACE (Ours)				
			Identity	K=4	K=32	K=128	Diffusion
GreatCourt	43/0.2	28/0.1	32/0.2	27/0.1	23/0.1	23/0.2	19/0.1
KingsCollege	28/0.4	18/0.3	30/0.4	18/0.3	19/0.3	22/0.4	19/0.3
OldHospital	31/0.6	25/0.5	34/0.6	21/0.4	20/0.4	21/0.4	17/0.4
ShopFacade	5/0.3	5/0.3	13/0.5	5/0.2	6/0.2	6/0.3	4/0.2
StMarysChurch	18/0.6	9/0.3	103/2.1	9/0.3	10/0.3	10/0.4	9/0.3
Average	25/0.4	17/0.3	42/0.8	16/0.3	15/0.3	17/0.3	14/0.3

Table 6. **Ablation of Global Encoding.** Performance of GLACE on the Cambridge Landmarks [24] with different kinds of global encoding input. We report median rotation and position errors.

we use K-Means to cluster the global encoding to certain discrete center values, we can explicitly force the grouping of the reprojection constraints. However, it is non-trivial to choose a suitable number of clusters, which may require a lot of tuning. In contrast, our feature diffusion technique achieves the best performance and additionally avoids tuning any hyperparameter.

Decoder. In Fig 6, we show the performance of our method with different numbers of decoder clusters K on the i19 dataset with SfM ground truth. When $K = 1$, which is equivalent to the original ACE [11] decoder, the network has an unimodal prior, which only learns the center scenes well and almost completely fails on several border scenes that are away from the center. When we increase the number of decoder clusters, the model is allowed to better parameterize a multimodal distribution and have increasing performance in border scenes. Note that different from the ensemble methods [11] that splits the scene into clusters and trains multiple models, our method of increasing the number of decoder clusters only needs to add a few output channels for the last linear layer and shows no significant increase in inference time and model size.

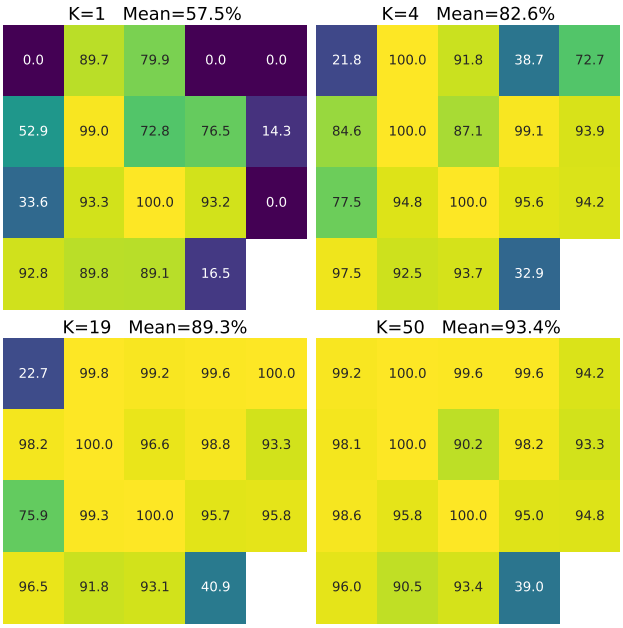

Figure 6. **Ablation of Decoder.** We compare the percentage of frames below a 5cm, 5° pose error for each room in the i19 integrated dataset.

5. Conclusion

In this paper, we have presented GLACE, a novel scene coordinate regression method that is able to work on large-scale scenes with a single network and without ground truth scene coordinate supervision. We propose a feature diffusion technique that effectively utilizes co-visibility information in the form of global encoding from image retrieval networks, to implicitly group the reprojection constraints and avoid overfitting to trivial solutions. We also propose a position decoder to effectively parameterize output coordinates in large-scale scenes. We believe that our insights and technical solutions are also applicable to other SCR methods to improve their performance on large-scale scenes.

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *CVPR*, 2016. 2

[2] Vassileios Balntas, Shuda Li, and Victor Prisacariu. RelocNet: Continuous metric learning relocalisation using neural nets. In *ECCV*, 2018. 2

[3] Daniel Barath, Jiri Matas, and Jana Noskova. MAGSAC: marginalizing sample consensus. In *CVPR*, 2019. 1

[4] Daniel Barath, Jana Noskova, Maksym Ivashechkin, and Jiri Matas. MAGSAC++, a fast, reliable and accurate robust estimator. In *CVPR*, 2020. 1

[5] Eric Brachmann and Carsten Rother. Learning Less is More - 6D Camera Localization via 3D Surface Regression. In *CVPR*, 2018. 1, 2, 3

[6] Eric Brachmann and Carsten Rother. Expert sample consensus applied to camera re-localization. In *ICCV*, 2019. 1, 2, 3, 6, 7, 8

[7] Eric Brachmann and Carsten Rother. Visual camera relocalization from RGB and RGB-D images using DSAC. *IEEE TPAMI*, 2021. 2, 6, 7

[8] Eric Brachmann, Frank Michel, Alexander Krull, Michael Y. Yang, Stefan Gumhold, and Carsten Rother. Uncertainty-driven 6D pose estimation of objects and scenes from a single RGB image. In *CVPR*, 2016. 2

[9] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. DSAC-differentiable ransac for camera localization. In *CVPR*, 2017. 1, 2, 3, 7

[10] Eric Brachmann, Martin Humenberger, Carsten Rother, and Torsten Sattler. On the limits of pseudo ground truth in visual camera re-localisation. In *ICCV*, 2021. 6

[11] Eric Brachmann, Tommaso Cavallari, and Victor Adrian Prisacariu. Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5044–5053, 2023. 1, 2, 3, 5, 6, 7, 8

[12] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-aware learning of maps for camera localization. In *CVPR*, 2018. 2

[13] Martin Bujnak, Zuzana Kukelova, and Tomas Pajdla. A general solution to the p4p problem for camera with unknown focal length. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. 1

[14] Federico Camposeco, Andrea Cohen, Marc Pollefeys, and Torsten Sattler. Hybrid scene compression for visual localization. In *CVPR*, 2019. 7

[15] Tommaso Cavallari, Stuart Golodetz, Nicholas A Lord, Julien Valentin, Luigi Di Stefano, and Philip HS Torr. On-the-fly adaptation of regression forests for online camera relocalisation. In *CVPR*, 2017. 2

[16] Tommaso Cavallari, Luca Bertinetto, Jishnu Mukhoti, Philip Torr, and Stuart Golodetz. Let's take this online: Adapting scene coordinate regression network predictions for online rgb-d camera relocalisation. In *3DV*, 2019. 1

[17] Tommaso Cavallari, Stuart Golodetz, Nicholas A. Lord, Julien Valentin, Victor A. Prisacariu, Luigi Di Stefano, and Philip H. S. Torr. Real-time rgb-d camera pose estimation in novel scenes using a relocalisation cascade. *TPAMI*, 2019. 2

[18] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPRW*, 2018. 6, 7, 8

[19] Mingyu Ding, Zhe Wang, Jiankai Sun, Jianping Shi, and Ping Luo. Camnet: Coarse-to-fine retrieval for camera relocalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2871–2880, 2019. 2

[20] Siyan Dong, Shuzhe Wang, Yixin Zhuang, Juho Kannala, Marc Pollefeys, and Baoquan Chen. Visual localization via few-shot scene region classification. In *3DV*, 2022. 1, 2, 6, 7

[21] Bert M Haralick, Chung-Nan Lee, Karsten Ottenberg, and Michael Nölle. Review and analysis of solutions of the three point perspective pose estimation problem. *International journal of computer vision*, 13:331–356, 1994. 1

[22] Martin Humenberger, Yohann Cabon, Nicolas Guerin, Julien Morat, Jérôme Revaud, Philippe Rerole, Noé Pion, Cesar de Souza, Vincent Leroy, and Gabriela Csurka. Robust image retrieval-based visual localization using Kapture, 2020. 6

[23] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. *CVPR*, 2017. 2, 7

[24] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-DOF camera relocalization. In *CVPR*, 2015. 1, 2, 6, 7, 8

[25] Xiaotian Li, Shuzhe Wang, Yi Zhao, Jakob Verbeek, and Juho Kannala. Hierarchical scene coordinate classification and regression for visual localization. In *CVPR*, 2020. 2, 3

[26] Tayyab Naseer and Wolfram Burgard. Deep regression for monocular camera-based 6-dof global localization in outdoor environments. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1525–1530. IEEE, 2017. 2

[27] Vojtech Panek, Zuzana Kukelova, and Torsten Sattler. MeshLoc: Mesh-Based Visual Localization. In *ECCV*, 2022. 2

[28] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *ICML*, 2019. 3

[29] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *CVPR*, 2019. 1, 2, 6, 7

[30] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*, 2020. 1, 2, 6, 7

[31] Paul-Edouard Sarlin, Ajaykumar Unagar, Måns Larsson, Hugo Germain, Carl Toft, Victor Larsson, Marc Pollefeys, Vincent Lepetit, Lars Hammarstrand, Fredrik Kahl, and Torsten Sattler. Back to the Feature: Learning Robust Camera Localization from Pixels to Pose. In *CVPR*, 2021. 6, 7

[32] Torsten Sattler, Tobias Weyand, Bastian Leibe, and Leif Kobbelt. Image retrieval for image-based localization revisited. In *BMVC*, 2012. 6, 8

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

[34] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1744–1756, 2016. 1, 2

[35] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & Effective Prioritized Matching for Large-Scale Image-Based Localization. *IEEE TPAMI*, 2017. 6, 7

[36] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, Fredrik Kahl, and Tomas Pajdla. Benchmarking 6DOF outdoor visual localization in changing conditions. In *CVPR*, 2018. 6, 8

[37] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixe. Understanding the limitations of cnn-based absolute camera pose regression. In *CVPR*, 2019. 5

[38] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1

[39] Johannes L Schönberger, Marc Pollefeys, Andreas Geiger, and Torsten Sattler. Semantic visual localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6896–6906, 2018. 1, 2

[40] Yoli Shavit, Ron Ferens, and Yosi Keller. Learning multi-scene absolute pose regression with transformers. In *ICCV*, 2021. 2, 7

[41] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in RGB-D images. In *CVPR*, 2013. 2, 6

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

[43] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7199–7209, 2018. 1, 2

[44] Hajime Taira, Ignacio Rocco, Jiri Sedlar, Masatoshi Okutomi, Josef Sivic, Tomas Pajdla, Torsten Sattler, and Akihiko Torii. Is this the right place? geometric-semantic pose verification for indoor visual localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4373–4383, 2019. 1, 2

[45] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *NeurIPS*, 2020. 3

[46] Carl Toft, Will Maddern, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, et al. Long-term visual localization revisited. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4):2074–2088, 2020. 1

[47] Akihiko Torii, Relja Arandjelovic, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 place recognition by view synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1808–1817, 2015. 2

[48] Mehmet Özgür Türkoğlu, Eric Brachmann, Konrad Schindler, Gabriel Brostow, and Áron Monszpart. Visual Camera Re-Localization Using Graph Neural Networks and Relative Pose Supervision. In *3DV*, 2021. 2

[49] Julien Valentin, Matthias Nießner, Jamie Shotton, Andrew Fitzgibbon, Shahram Izadi, and Philip H. S. Torr. Exploiting uncertainty in regression forests for accurate camera relocalization. In *CVPR*, 2015. 2

[50] Julien Valentin, Angela Dai, Matthias Nießner, Pushmeet Kohli, Philip Torr, Shahram Izadi, and Cem Keskin. Learning to navigate the energy landscape. In *3DV*, 2016. 2, 6

[51] Florian Walch, Caner Hazirbas, Laura Leal-Taixé, Torsten Sattler, Sebastian Hilsenbeck, and Daniel Cremers. Image-based localization using LSTMs for structured feature correlation. In *ICCV*, 2017. 2

[52] Shuzhe Wang, Zakaria Laskar, Iaroslav Melekhov, Xiaotian Li, Yi Zhao, Giorgos Tolias, and Juho Kannala. Hscnet++: Hierarchical scene coordinate classification and regression for visual localization with transformer. *CoRR*, abs/2305.03595, 2023. 3, 6

[53] Frederik Warburg, Søren Hauberg, Manuel López-Antequera, Pau Gargallo, Yubin Kuang, and Javier Civera. Mapillary street-level sequences: A dataset for lifelong place recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 4

[54] Dominik Winkelbauer, Maximilian Denninger, and Rudolph Triebel. Learning to localize in new environments from synthetic training data. In *ICRA*, 2021. 2

[55] Luwei Yang, Ziqian Bai, Chengzhou Tang, Honghua Li, Yasutaka Furukawa, and Ping Tan. SANet: Scene agnostic network for camera localization. In *ICCV*, 2019. 6, 7

[56] Qunjie Zhou, Torsten Sattler, Marc Pollefeys, and Laura Leal-Taixe. To learn or not to learn: Visual localization from essential matrices. In *ICRA*, 2020. 2

[57] Qunjie Zhou, Sérgio Agostinho, Aljoša Ošep, and Laura Leal-Taixé. Is geometry enough for matching in visual localization? In *ECCV*, 2022. 2, 7

[58] Sijie Zhu, Linjie Yang, Chen Chen, Mubarak Shah, Xiaohui Shen, and Heng Wang. R^2former: Unified retrieval and reranking transformer for place recognition. *CoRR*, abs/2304.03410, 2023. 3, 4