SegLoc: Learning Segmentation-based Representations for Privacy-Preserving Visual Localization

Maxime Pietrantoni\textsuperscript{1,2}, Martin Humenberger\textsuperscript{3}, Torsten Sattler\textsuperscript{2}, Gabriela Csurka\textsuperscript{3}

\textsuperscript{1}Faculty of Electrical Engineering, Czech Technical University in Prague
\textsuperscript{2}Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague
\textsuperscript{3}NAVER LABS Europe

\{firstname.lastname\}@cvut.cz,\{firstname.lastname\}@naverlabs.com

Abstract

Inspired by properties of semantic segmentation, in this paper we investigate how to leverage robust image segmentation in the context of privacy-preserving visual localization. We propose a new localization framework, SegLoc, that leverages image segmentation to create robust, compact, and privacy-preserving scene representations, i.e., 3D maps. We build upon the correspondence-supervised, fine-grained segmentation approach from \cite{42}, making it more robust by learning a set of cluster labels with discriminative clustering, additional consistency regularization terms and we jointly learn a global image representation along with a dense local representation. In our localization pipeline, the former will be used for retrieving the most similar images, the latter to refine the retrieved poses by minimizing the label inconsistency between the 3D points of the map and their projection onto the query image. In various experiments, we show that our proposed representation allows to achieve (close-to) state-of-the-art pose estimation results while only using a compact 3D map that does not contain enough information about the original images for an attacker to reconstruct personal information.

1. Introduction

Visual localization is the problem of estimating the precise camera pose — position and orientation — from which the image was taken in a known scene. It is a core component of systems such as self-driving cars \cite{31}, autonomous robots \cite{49}, and mixed-reality applications \cite{4,53}.

Traditionally, visual localization algorithms rely on a 3D scene representation of the target area, which can be a 3D point cloud map \cite{29,34,35,45,46,66,68,69,73,79}, e.g., from Structure-from-Motion (SfM), or a learned 3D representation \cite{9,10,14,37,38,71,76}. The representation is typically derived from reference images with known camera poses. Depending on the application scenario, these maps need to be stored in the cloud, which raises important questions about memory consumption and privacy preservation. It is possible to reconstruct images from maps that contain local image features \cite{62}, amongst the most widely used for scene representation.

To tackle the above challenges that feature-based approaches may face, inspired by semantic-based \cite{48,82} and segmentation-based \cite{42} approaches, we propose a visual localization pipeline where robust segmentations are used as the sole cue for localization, yielding reduced storage requirements (compared to using local features) while increasing privacy-preservation. Our proposed localization pipeline, called SegLoc, follows standard structure based-localization pipelines \cite{34,66} that represent the scene via a 3D model: first, image retrieval based on a compact image representation is used to coarsely localize a query image. Given such an initial pose estimate, the camera pose is refined by aligning the query image to the 3D map. Contrary to prior work that is based on extracting features directly from images, we derive a more abstract representation in the form of a robust dense segmentation based on a...
set of clusters learned in a self-supervised manner. As illustrated in Figure 1, we use this segmentation to both extract a global descriptor for image retrieval and for pose refinement. The pose is refined by maximizing the label consistency between the predictions in the query image and a set of labeled 3D points in the scene.

Such an approach has multiple advantages. First, our model is able to learn representations which are robust to seasonal or appearance changes. Similar to semantic segmentations, which are invariant to viewing conditions as the semantic meaning of regions do not change, our representation is trained such that the same 3D point is mapped to the same label regardless of viewing conditions. Second, it results in low storage requirements, as instead of storing high-dimensional feature descriptors, for each 3D point we only keep its label. Finally, it allows privacy-preserving visual localization [15, 22, 28, 78], as it creates a non-injective mapping from multiple images showing similar objects with different appearances to similar labels. While, ensuring user privacy comes at the cost of reduced pose accuracy [19, 98], our method comes close to state-of-the-art results with a better accuracy vs. memory vs. privacy trade-off.

To summarize, our first contribution is a new localization framework, called SegLoc, that extends the idea [41, 42] of learning robust fine-grained image segmentations in a self-supervised manner. To that end, we leverage discriminative clustering while putting more emphasis on representation learning. Furthermore, we derive a full localization pipeline, where our model jointly learns global image representation to retrieve images for pose initialization, and dense local representations for building a compact 3D map – an order of magnitude smaller compared to feature-based approaches – and to perform privacy-preserving pose refinement. As a second contribution, we draw a connection between segmentation-based representations and privacy-preserving localization, opening up viable alternatives to keypoint-based methods within the accuracy-privacy-memory trade-off. We evaluate our approach in multiple indoor and outdoor environments while quantitatively measuring privacy through detailed experiments.

2. Related Work

Semantic-based Visual Localization. Semantic segmentation is used in structure-based localization methods as a way to facilitate feature selection or matching [2, 39, 40, 54, 56, 72], to filter 2D-3D matches by maximizing the semantic consistency between 2D images and 3D models [42, 74, 83] or to improve keypoint tracking [48]. In these works, the pre-trained segmentation model is used mainly to filter matches or to improve SfM/VO, hence they still rely on keypoints descriptors. Similar to FGSN [42], our model learns robust image segmentation in a self-supervised manner exploiting label consistency between matched keypoints. Contrary to FGSN, our model provides both global and local representations, resulting in a full localization pipeline.

Semantics-based retrieval and pose approximation. To cope with extreme environmental, seasonal, and illumination changes in place recognition and image retrieval, several methods leverage image-to-image translation to handle the domain shift between the database and query images [1, 32, 33, 63, 80, 96, 97]. Other methods directly aim at obtaining image representation by leveraging weather-invariant semantic [7, 26, 82, 94] or geometric information [59, 60]. In particular, [82] describes images through histograms of semantic classes from pre-trained semantic segmentation, while LoST [26] performs a semantic-based pooling of convolutional features. Closest to our work, [32] and [58] train global representations within a deep metric learning framework and utilize semantic segmentation as an auxiliary task to infuse semantic information. Instead, we learn a finite set of (not necessarily semantic) classes to perform image segmentation from which we build local and global representations.

Pose refinement. Pose refinement approaches obtain an accurate camera pose estimate from an initial approximate pose via image alignment. In contrast to early methods based on handcrafted features [55] or pixel intensities [23], more recent methods learn deep features through direct feature alignment suitable for such pose refinement [67, 87] or cast the camera pose localization as a metric learning problem. [82] proposed a semantic-based pose refinement relying on a pre-trained model, handcrafted global descriptors, and a geometric prior. Our pose refinement is inspired by PixLoc [67], except that instead of using multi-scale deep features, we align 1D features (labels) by minimizing a re-projection error as a function of label inconsistency.

Privacy-preserving localization. Storing the 3D maps on the cloud or sending images or descriptors from mobile devices to a server raise the question of privacy. As shown in [62], detailed and recognizable images of the scene can be obtained from sparse 3D point clouds with local descriptors. Geometry-based matching methods [12, 98] do not rely on visual descriptors to localize, hence they are less subject to privacy issues. Still, [62] shows that depth and color are sometimes sufficient to recover details in a scene. Learning-based pose regression or scene point regression models [8, 38, 88, 91] do not explicitly store the 3D map and, thus, partially avoid the privacy issues. Yet, according to [51], since these models memorize the scenes quite well, model inversion is often possible. Given a set of pre-selected scene landmarks, [19] learns to detect them and to regress the associated bearing vectors used by geometric camera pose estimation. However, this method still faces the same scaling issues as regression methods.

To address privacy, [75, 77] propose to transform 3D point clouds into 3D line clouds, thus obfuscating the scene
geometry. However, according to [15], a significant amount of information about the scene geometry is preserved in these line clouds, allowing to (approximately) recover image content. [28] propose a cloud-based mapping solution to preserve the privacy of users by hiding critical content of the input images. As the recovered pose may also be considered as sensitive, [27] perform a partial estimation of a 3DoF pose on a single dimension against a partial map. These partials maps are distributed on distinct servers so that the 6DoF pose can only be recovered on the user side. However, they do not tackle the privacy of the partial maps directly. In addition, these approaches do not consider private information contained in the query images, which could, e.g., allow an attacker to track individuals and to study their behavior. Concerning privacy preservation of the query, [20,21] show that it is possible to reconstruct the original image from local image features. To address this, [22] propose to obfuscate the appearance of the original image by lifting the descriptors to affine subspaces. [78] propose to replace 2D points in the query image with randomly oriented 2D lines passing through the given point. This allows them to address privacy of both the query and the map. By relying on class labels, where only labels are kept in the map and hence making it impossible to recover fine details that could reveal private information, our SegLoc representations jointly tackles query and map privacy.

3. The SegLoc Model

Our goal is to jointly learn local and global representations for visual localization. Inspired by the invariance of semantic class labels to viewing conditions, we propose a robust image segmentation method based on a set of clusters uncovered in a self-supervised manner. To make the segmentation robust to viewpoint and appearance changes, we train our model on an ensemble of image pairs taken from different viewpoints and at different points in time with a set of automatically extracted keypoint correspondences between them (see Supplementary). We assume a pre-trained encoder providing initial dense representations, which are grouped into $K$ prototypes, where $K$ is the number of clusters / labels / classes representing the desired segmentation granularity. They are used to initialize the segmentation heads and the discriminative clustering step.

Hence, the main ingredients of our model are: dense segmentation as representations learned with discriminative clustering (Sec. 3.1), three additional consistency regularization terms (Sec. 3.2), and global image representations trained with a multi-similarity pairwise loss (Sec. 3.3).

3.1. Multi-scale dense representation

The segmentation network has a hierarchical structure and uses a hybrid-DPT [64] like encoder-decoder module as backbone such that the output of each level is the input of the next one. The resolutions of the output decoded feature maps $F_l \in \mathbb{R}^{D \times H_l \times W_l}$ progressively increases. Each feature map $F_l$ is further processed by a classification head $h_{ul}$ parametrized by $\mu_l$ which predicts a set of yielding segmentation heatmaps $P^l_k \in \mathbb{R}^{H_l \times W_l}$ – with per pixel class likelihoods – corresponding the $k^{th}$ cluster. The tensor concatenating the $K$ maps, denoted by $P_1 \in \mathbb{R}^{K \times H_l \times W_l}$, is an abstract representation of the image. As the decoder outputs higher resolution feature maps the encoded information becomes finer. We thus operate on four complementary distinct metric spaces and classification spaces ($l \in \{1,\ldots,4\}$, learning four distinct cluster sets, one per level. During pose refinement, we hierarchically use these maps from coarser to finer to leverage information from different levels of granularity. For pose approximation only the finer segmentation is used to compute the global representation while we use the four segmentations for pose refinement. In the following, we drop the level notation $l$ as the described steps are applied on each level without distinction.

**Discriminative clustering.** For clustering, we rely on a Deep Discriminative Clustering (DDC) framework [18, 36, 92] as it focuses on learning the boundaries between clusters rather than explicitly modeling the data distribution, hence casting the clustering task as a classification problem. Following [18], we use an auxiliary target to supervise the training by minimizing the Kullback-Leibler (KL) divergence between the predicted distributions $P$ and target distributions $Q$. To avoid degenerated solutions, [18] use a regularization term that minimizes $KL(d^q || d^p)$ between the empirical label distribution $d^q$ defined as the soft frequency of cluster assignments in the target distribution and the uniform distribution $d^u$ to enforce a balanced cluster assignments. We instead rely on the data itself to directly estimate an empirical label distribution $d^p$.

We add an entropy term $H(Q)$ that encourages peaked target distributions and minimize the clustering objective:

$$L_{DC} = KL(Q||P) + KL(d^q || d^p) + H(Q),$$

where $d^q_k = \sum_i^{HW} q_{ik}$ and $B$ is the batch size. As this objective depends both on the target distributions $Q$ and the network parameters, it is minimized by alternating the following two sub-steps in every batch:

1. **Update target distribution:** With network parameters fixed, the following closed-form solution minimizes the cost function Eq. (1) in a batch of size $B$:

$$q_{ik} = \frac{d^p_k P_{ik}/(\sum_{k'}^{B} \sum_{i,j=1}^{HW} P^i_{jk'})^{\frac{1}{2}}}{\sum_{k=1}^{K} d^p_k P_{ik}/(\sum_{k'}^{B} \sum_{i,j=1}^{HW} P^i_{jk'})^{\frac{1}{2}}}.$$

Enforcing uniform prior is not desirable as information is unbalanced in the dataset and even more in a batch.
2. Update model parameters: With target distributions fixed, minimizing the cost function accounts to minimizing the following per-pixel cross entropy loss:

\[ \mathcal{L}_{CE} = - \frac{1}{HWB} \sum_{b=1}^{B} \sum_{i=1}^{HW} \sum_{k=1}^{K} q_{ik} \log(p_{ik}) \, , \tag{3} \]

with \( \sigma \) being the softmax function.

The model is self-supervised by the auxiliary target distributions \( Q \), where \( q_{ik} \) are computed from the initial class predictions \( p_{ik} \). However, these predictions are not reliable at the beginning of the training process. Therefore, during the first epoch, instead of using Eq. (2) to update \( Q \), we rely on some initial prototypes (cluster centers) \( c_k \).

Specifically, we compute soft class assignments w.r.t. the associated clusters for each pixel \( x_i \) using a Student’s t-distribution \([86]\):

\[ q_{ik} = \frac{(1 + ||F_i - c_k||^2 / \alpha)^{-\frac{1+b}{2}}}{\sum_{k'=1}^{K}(1 + ||F_i - c_{k'}||^2 / \alpha)^{-\frac{1+b}{2}}} \, , \tag{4} \]

using the corresponding feature vectors \( F_i \) and \( \alpha = 1 \) as in \([92]\). Using Eq. (4) instead of Eq. (2) in the first epoch acts not only as initialization, but also allows to distill underlying prior knowledge (see details in the Supplementary), helping the learning process to be more efficient.

3.2. Consistency regularization

Aiming to learn dense segmentations robust to photometric changes while being equivariant to viewpoint changes and to avoid overfitting, we propose the following three consistency regularization losses.

Formally, let \( I^a, I^b \) be an image pair with the corresponding 2D-normalized feature maps \( F^a = f_b(I^a) \) and \( F^b = f_b(I^b) \) respectively. We denote the set of automatically obtained 2D-2D keypoint correspondences by \( \{x_{u_i}^a, x_{v_i}^b\}_{i=1}^L \), where \( x_{u_i}^a \) and \( x_{v_i}^b \) are the keypoint locations in the feature maps \( F^a \) respectively \( F^b \). We define the following losses:

**Correspondence consistency loss.** Similar to \([42]\), to enforce consistency between pairs of segmentations we use a correspondence consistency loss:

\[ \mathcal{L}_{CC} = - \frac{1}{2L} \sum_{l=1}^L \mathbb{1}_{s_{u_l}^a} \log(\sigma(p_{u_l}^a)) + \mathbb{1}_{s_{v_l}^b} \log(\sigma(p_{v_l}^b)) \, , \]

where \( p_{u_l}^a = h_{\mu_k(F_{u_l})} \), \( p_{v_l}^b = h_{\mu_k(F_{v_l})} \), \( l_k \) is the one-hot vector with all zero values except at position \( k \), \( s_{u_l}^a \) and \( s_{v_l}^b \) are the hard-assigned cluster labels to the keypoints \( x_{u_l}^a \) and \( x_{v_l}^b \) based on their distance to the prototypes \( \{c_k\}_{k=1}^K \) based on: \( s_{u_l}^a = \arg\max_{k} c_k^\top F_{u_l} \) and \( s_{v_l}^b = \arg\max_{k} c_k^\top F_{v_l} \).

**Prototypical cross contrastive loss.** To constrain the feature space to ensure separability between the implicitly defined classes and to improve intra-class compactness, we define the prototypical cross contrastive loss, inspired by the ProtoNCE \([44]\), as follows:

\[ \mathcal{L}_{PC} = - \frac{1}{2L} \sum_{l=1}^L \log \left( \frac{1}{Z} \exp \left( \frac{c_{u_l}^\top F_{u_l}}{\phi_{e_{u_l}}} + \frac{c_{v_l}^\top F_{v_l}}{\phi_{e_{v_l}}} \right) \right) \, , \]

with \( Z = (\sum_k \exp (c_{u_l}^\top F_{u_l}/\phi_k)) (\sum_k \exp (c_{v_l}^\top F_{v_l}/\phi_k)) \), \( \phi_k \) being the concentration of the prototype \( c_k \) defined as the average feature distance to the prototype within the cluster \( k \) and it acts as a scaling factor preventing cluster collapse. This loss incorporates in the feature space the structure conveyed by the prototypes.

**Contrastive feature consistency loss.** To exploit the relationships between keypoints, we adapt the supervised contrastive loss \([85]\) to enforce feature consistency:

\[ \mathcal{L}_{FC} = - \frac{1}{L} \sum_{l=1}^L \log \frac{\exp(F_{u_l}^a \top F_{v_l}^b / \tau)}{\sum_{l'=1}^L \exp(F_{u_l'}^a \top F_{v_l}^b / \tau)} \, . \tag{5} \]

The anchor/positive pairs are provided by the pixel-to-pixel correspondences, while the negatives are obtained by sampling amongst the other keypoints in the set \( \{x_{u_l}^a, j \neq l\} \). This loss forces the features of corresponding keypoints to be similar, hence facilitating the subsequent clustering.

3.3. Segmentation-based global descriptor

To fully leverage our segmentations, we propose to compute a global image representation by applying a pooling operator on the segmentation heatmap instead of the feature maps. We use the Generalized Pooling Operator (GPO) \([16]\), which generalizes over different pooling strategies to learn the most appropriate pooling strategy to describe the content. Given a heatmap’s channel \( P^k \in \mathbb{R}^{HW} \), it is defined as a weighted sum over sorted features:

\[ v^k = \sum_{o=1}^{HW} \theta_o \psi^k_o \quad \text{where} \quad \sum_{o=1}^{HW} \theta_o = 1 \, , \tag{6} \]

where \( v^k \) is the \( k \)th element of the output feature vector, \( \psi^k_o \) is the \( o \)th element from the ordered descending lists of the values in the in the heatmap’s channel; \( P^k \) and the weights \( \theta_o \) are shared between the channels. We use the higher resolution heatmap from the last level of the decoder as input. The segmentation labels provide a much weaker signal compared to features, we thus opted for spatial pooling to increase discriminativeness instead of using a permutation-invariant pooling strategy such as \([3]\). We divide the image into \( M \) overlapping sliding sub-windows and apply pooling within each sub-window. The corresponding features are then concatenated yielding a global representation of dimension \( MK \). While this implies lower robustness to viewpoint change, in practice we find it sufficient since subsequent pose refinement requires an initial pose close enough.
to the true pose to enable convergence (c.f. Fig. A.5 in the Supplementary). PCA-whitening postprocessing is applied to reduce the dimension to 4096.

**Multi-similarity loss.** We consider as global training objective the multi-similarity loss [90]. Given an anchor image \( I^a_j \), we denote the corresponding positive respectively negative image sets by \( \mathcal{N}^+_m = \{ I^a_j \} \) and \( \mathcal{N}^-_m = \{ I^a_j \} \) and the corresponding similarities, computed between the pooled global representations by \( S^+_m \) and \( S^-_m \). The multi-similarity loss is then defined as:

\[
L_{MS} = \frac{1}{N} \sum_{n=1}^{N} \sum_{\rho \in \{+, -\}} \frac{1}{\alpha^\rho} \log \left( 1 + \sum_{I^a_j \in \mathcal{N}^\rho_m} e^{\rho \alpha^\rho \rho (\lambda - S^\rho_m)} \right)
\]

where \( \alpha^+, \alpha^- \) and \( \lambda \) are hyper-parameters. We use image pairs contained in our dataset as anchor/positive pair. According to standard practices, the rest of positive/negative samples are mined from \( I^a_n, \mathcal{N}^\rho_m \) through a semi-hard mining scheme based [24, 30] on features distances and image positions (see details in the Supplementary).

**5. Experimental Evaluation**

**Training and test data.** We train and evaluate our model in both indoor and outdoor scenes. Outdoors, we created an extended version of the Cross-Seasons Correspondences dataset [41] (built upon the training slices of ECMU [5, 81]), including more diverse intra-seasons image pairs and larger viewpoint changes between image pairs (see Supplementary). Indoors, we use the challenging Indoor6 dataset [19] from which we sample pairs of co-visible images captured under different conditions and compute their correspondences based on geometry. For in-domain evaluation, we use the test sets of ECMU and Indoor6, for evaluating the generalization ability, we use RobotCar Seasons (RC) [50, 70] and Cambridge Landmarks [38]. Our models use 100 classes throughout the whole experimental section.

**Evaluation protocol.** To measure pose accuracy, we follow [61, 70] and compute the position and rotation errors between the estimated query pose and the ground truth pose. We report the percentage of query images localized within fine (.25m, 2°), medium (.5m, 5°) and coarse (5m, 10°) thresholds for outdoor environments and median translation and rotation errors in (cm/°) as well as the localization recall at (5cm, 5°) for indoor environments.

**5.1. Results and ablative study**

**Pose approximation results.** Tab. 1 compares our global descriptor against four popular global representations used for localization [61], DELG [13], APGeM [65], DenSeVLAD [84] and NetVLAD [3]. Additionally, we include global descriptors that implicitly leverage semantic information: DASGIL [32], DIFL-FCL [33] and LVLPR [93]. On ECMU, our global descriptor significantly outperforms all existing representations, demonstrating the discriminativeness of the learned segmentations. Furthermore, SegLoc performs very well in day conditions of the RC dataset, despite not being trained on it. The drop in performance between day and night images on RC can be explained by the fact that our training set does not include nighttime photos. Still, SegLoc is only outperformed by LVLPR on the coarser thresholds, which was trained on RC.

**Pose refinement results.** Tab. 2 compares our pose refinement approach against PixLoc [67] (also trained on
Table 1. Pose approximation (PA) results obtained with the pose of the top-1 retrieved images using different global representations.

<table>
<thead>
<tr>
<th>Model</th>
<th>Trained on</th>
<th>ECMU Seasons</th>
<th>RC Seasons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Urban</td>
<td>Suburban</td>
<td>Park</td>
</tr>
<tr>
<td>DELG [17]</td>
<td>GL18 [57]</td>
<td>7.8 / 19.7 / 73.7</td>
<td>2.5 / 9.6 / 66.2</td>
</tr>
<tr>
<td>AP GeM [65]</td>
<td>GL18 [57]</td>
<td>8.0 / 20.6 / 74.7</td>
<td>2.7 / 10.0 / 63.8</td>
</tr>
<tr>
<td>DenseVLAD [64]</td>
<td>247 Tokyo [64]</td>
<td>12.2 / 29.2 / 74.7</td>
<td>4.7 / 17.3 / 73.0</td>
</tr>
<tr>
<td>DFL-FCL [33]</td>
<td>ECMU [58,81]</td>
<td>14.8 / 35.1 / 79.6</td>
<td>5.6 / 18.2 / 69.8</td>
</tr>
<tr>
<td>LVPR [83]</td>
<td>RC [50,70]</td>
<td>17.3 / 42.5 / 89.0</td>
<td>5.8 / 19.4 / 76.1</td>
</tr>
<tr>
<td>DASGIL [32]</td>
<td>Virtual KITTI [25]</td>
<td>17.4 / 42.0 / 91.1</td>
<td>6.7 / 22.1 / 88.5</td>
</tr>
<tr>
<td>SegLoc</td>
<td>ECMU [58,81]</td>
<td>21.5 / 51.7 / 96.5</td>
<td>8.7 / 28.5 / 92.6</td>
</tr>
</tbody>
</table>

Table 2. Comparing cluster-based SegLoc with feature-based PixLoc (both trained on ECMU) on the pose refinement (PR) task in terms of pose accuracy, memory requirements, and privacy of the underlying 3D map representation. Privacy is evaluated by recovering images from the point clouds used by both methods (worse image quality implies a higher level of privacy). For a better comparison we also evaluate SegLoc using NetVLAD for retrieval (NV).

<table>
<thead>
<tr>
<th>Model</th>
<th>Memory (GB)</th>
<th>PSNR (↑)</th>
<th>Reconstruction quality</th>
<th>Localization Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>LPIPS ↓</td>
<td>SSIM ↑</td>
<td>MAE ↓</td>
</tr>
<tr>
<td>SegLoc NV</td>
<td>0.102</td>
<td>0.06</td>
<td>0.211</td>
<td>0.14</td>
</tr>
<tr>
<td>SegLoc 1L</td>
<td>0.156</td>
<td>0.46</td>
<td>0.211</td>
<td>0.14</td>
</tr>
<tr>
<td>SegLoc NV</td>
<td>0.102</td>
<td>0.06</td>
<td>0.211</td>
<td>0.14</td>
</tr>
<tr>
<td>SegLoc 1L</td>
<td>0.156</td>
<td>0.46</td>
<td>0.211</td>
<td>0.14</td>
</tr>
</tbody>
</table>

ECMU), a state-of-the-art pose refinement method. We report pose accuracy results for ECMU and RC together with the storage requirements of the two methods. We further analyze both approaches in terms of map privacy. To this end, we train the map inversion approach from [22, 62] to recover images from the point clouds used by both PixLoc and our approach (details in Sec. 5.2). Comparing the recovered images with the original ones, we report the PSNR, LPIPS [95], SSIM, and MAE metrics. Low scores are desired for PSNR and SSIM (resp. high scores for LPIPS and MAE) as this means the images cannot be well reconstructed. For visualizations and a memory consumption report, see Supplementary. The results verify our claim that our approach is more privacy-preserving than feature-based methods as the image reconstruction results are significantly worse (see also Sec. 5.2). Furthermore, our approach requires significantly less storage space.

Concerning accuracy, using solely SegLoc as a full pipeline (using a single model to compute representations for both retrieval and pose refinement) outperforms PixLoc on ECMU. Particularly in the "suburban" and "park" scenes, where it is hard to find stable and reliable local features in scenes dominated by vegetation – especially under seasonal changes, our representations are more robust. In the case of "park", SegLoc even outperforms PixLoc on coarser accuracy levels, even if PixLoc uses an "oracle" ranking. These gains are partially due to our robust global representation. When using NetVLAD to initialize our poses (SegLoc NV), our results are below the PixLoc performance. This comes at no surprise given that PixLoc uses high-dimensional features that store significantly more information (see Tab. 2), confirming the observation made in [19, 98] that privacy-preservation comes at the cost of decreased pose accuracy.

With a limited semantic gap as in RC (or Cambridge Landmarks) with regard to the ECMU training set, our approach is still able to significantly refine initial poses as shown in Tab. 2 and Tab. 4. However, our approach is data-driven and uncovers a set of clusters without human supervision or dense annotations. While being somewhat interpretable, these clusters remain tied to the semantic space of the training dataset. Without explicit domain adaptation, we thus cannot expect strong generalization capabilities. This explains the gap between SegLoc and PixLoc results on RC Seasons. Improving generalization, e.g., by training on more data, is an interesting direction for future work.

Comparison to privacy preserving methods. Next, we compare against recent privacy-preserving visual localization methods, DSAC* [11], GoMatch [98], and NBE+SLD [19] on the Indoor6 [19] (Tab. 3, all methods are also trained on Indoor6 dataset) and Cambridge Landmarks [38] (Tab. 4). By design these methods do not scale to large outdoor environments, so we did not include them in our outdoor comparisons (ECMU and RC Seasons). On Indoor6, SegLoc significantly outperforms DSAC*, but in some
Table 3. Localizations results on Indoor6 compared in terms of memory footprint (required to store the map), and localization accuracy.

<table>
<thead>
<tr>
<th>Model</th>
<th>MB</th>
<th>King’s</th>
<th>Old</th>
<th>Shop</th>
<th>St. Mary’s</th>
</tr>
</thead>
<tbody>
<tr>
<td>GoMatch ✓ [98]</td>
<td>48</td>
<td>0.25/0.64</td>
<td>3.83/1.84</td>
<td>0.48/0.77</td>
<td>3.35/0.94</td>
</tr>
<tr>
<td>SegLoc ✓</td>
<td>23</td>
<td>0.24/0.26</td>
<td>0.36/0.52</td>
<td>0.11/0.34</td>
<td>0.17/0.46</td>
</tr>
<tr>
<td>PixLoc [67]</td>
<td>3545</td>
<td>0.14/0.24</td>
<td>0.16/0.32</td>
<td>0.05/0.23</td>
<td>0.10/0.34</td>
</tr>
</tbody>
</table>

Table 4. Comparison of median position and orientation errors (m./°) on Cambridge Landmarks [38]. We outperform the privacy-preserving GoMatch [98] approach in all metrics. For reference, we include the non-privacy-preserving PixLoc method, which requires more than two orders of magnitude more memory.

| DPT Hybrid [64] | 4.7/14/30/82.2 | 3.9/14/30/87.3 | 3.4/30/95.0 |
| SegLoc          | 26.8/51.52/3.3 | 15.1/32.18/6.8 | 10.3/22.8/6.0 |
| FGSN [42]       | 43.4/63.49/2.6 | 27.4/42/0.69/1 | 17.0/31/0/65.7 |

Table 5. Pose refinement results on ECMU when varying the segmentation models.

<table>
<thead>
<tr>
<th>Indoor</th>
<th>Outdoor</th>
</tr>
</thead>
<tbody>
<tr>
<td>SegLoc full</td>
<td>60.3</td>
</tr>
<tr>
<td>SegLoc top10</td>
<td>66.5</td>
</tr>
<tr>
<td>SegLoc full</td>
<td>66.5</td>
</tr>
<tr>
<td>Pixloc</td>
<td>76.3</td>
</tr>
<tr>
<td>Recall (5cm,5°) (%)</td>
<td>37.01</td>
</tr>
<tr>
<td>PSNR (dB)</td>
<td>14.44</td>
</tr>
<tr>
<td>LPIPS (↑)</td>
<td>0.32</td>
</tr>
<tr>
<td>SSIM (↑)</td>
<td>0.49</td>
</tr>
<tr>
<td>MAE (↑)</td>
<td>0.09</td>
</tr>
<tr>
<td>Indoor</td>
<td>33.4</td>
</tr>
<tr>
<td>SegLoc</td>
<td>60.3</td>
</tr>
<tr>
<td>SegLoc top10</td>
<td>66.5</td>
</tr>
<tr>
<td>SegLoc full</td>
<td>66.5</td>
</tr>
<tr>
<td>Pixloc</td>
<td>76.3</td>
</tr>
<tr>
<td>Recall (5cm,5°) (%)</td>
<td>37.01</td>
</tr>
<tr>
<td>PSNR (dB)</td>
<td>14.44</td>
</tr>
<tr>
<td>LPIPS (↑)</td>
<td>0.32</td>
</tr>
<tr>
<td>SSIM (↑)</td>
<td>0.49</td>
</tr>
<tr>
<td>MAE (↑)</td>
<td>0.09</td>
</tr>
</tbody>
</table>

Table 6. Ablation study for pose approximation (PA) and pose refinement (PR). We disable different losses and used an input size of 480x480 pixels and one feature level during training.

Table 7. Detection of privacy sensitive classes on reconstructed images (ECMU / Indoor6 test sets).

5.2. Accuracy vs privacy trade-off

As seen in our experiments above, and as already mentioned in [19, 98], ensuring user privacy comes at the cost of reduced pose accuracy. In this section, we explore this trade-off in more details. Inspired by [17], we quantify how privacy-preserving our approach is in comparison to PixLoc (the most similar in terms of pose refinement).

For any visual localization service, the user sends a query to a server and the latter performs visual localization against a stored database model and returns the 6DoF pose to the user. In this context, we define privacy as the inability for an attacker to recover critical details of the scene either from the query or the database. Qualitatively, we measure this degree of privacy through the output of feature/SIM model inversion approaches [22, 62] that recover images from 3D models or image representations. Quantitatively, we measure privacy using image reconstruction metrics and by quantifying the ability of detecting privacy-sensitive objects from the recovered images using an object detector.

Recovering map images. Given the reference 3D models
of ECMU, RC, and Indoor6, we train an inversion model [62] per dataset to recover images from these sparse SfM models. We learn inversion models for both SfM models where a 3D point is associated to a PixLoc descriptor or to a single SegLoc cluster label. We evaluate the models on reference 3D models of testing slices (unseen during training). Reconstruction metrics reported in Tab. 2 and visualizations provided in Fig. 2 show that SegLoc is both qualitatively and quantitatively more privacy-preserving than PixLoc (see e.g. in the top right example the reconstructed buildings and white car from PixLoc features).

Detecting sensitive areas. Reconstruction quality metrics are image level and do not evaluate what happens for particular objects. Therefore, we evaluate also discernability of sensitive classes (pedestrians, cars, indoor furniture) in the reconstructed images. To that end, we first evaluate the yolov7 [89] object detector on the original database images of ECMU and Indoor6. We use these detections as ground truth and try to detect the same classes from the reconstructed images. IoU metrics are reported in Tab. 7 and the corresponding bounding boxes shown in Fig. 2. While reconstructed images from SegLoc maps preserve the overall structure (which is encoded in the boundaries between classes), they do not contain recognizable details. On the contrary, PixLoc’s maps allows the detector to recover fine details on previously unseen images. Note furthermore that even when an object is “reconstructed” in an image, the details such as color or brand is not discovered (see e.g. the white car reconstructed as a dark one in Fig. 2 top row).

The privacy of the query. Finally, some scenarios might require that the query sent to the server be privacy preserving. As query, SegLoc can either use the dense segmentation, part of it, or a single label representation while PixLoc uses the dense feature map. Given ECMU and Indoor6 database images, we train a dense inversion model adapted from [62] to invert the aforementioned input representations. In Tab. 8, we report reconstruction results and associated localization performances for both ECMU and Indoor6 testing sets. Using a one-hot query guarantees a high level of privacy while increasing the amount of encoded information facilitates localization at the cost of lowering privacy.

6. Conclusion

This paper explored to what extent robust segmentations based on a set of clusters may be used as an alternative intermediate representation for visual localization. Given the increasing concerns about privacy and storage requirements, such representations promise a competitive discriminativeness-privacy-memory trade-off. To address this, we proposed a novel method SegLoc that jointly learns global image descriptor and dense local representations by uncovering underlying clusters in a weakly-supervised manner. Such classes enable 3D representations of the environment that are an order of magnitude lighter than feature descriptor-based 3D maps. Despite the loss of information induced by using segmentations with a finite number of classes, we show that our method comes close to the performance of state-of-the-art feature based-methods on outdoor and indoor environments. Furthermore, we explicitly establish a connection between robust segmentation-based localization and privacy-preserving localization, showing that our representations offer an excellent trade-off between pose accuracy, privacy preservation, and memory requirements, opening new perspectives for visual localization.

Acknowledgments. Maxime and Torsten received funding from NAVER LABS Europe.
References


[80] Li Tang, Yue Wang, Qianhui Luo, Xiaqing Ding, and Rong Xiong. Adversarial Feature Disentanglement for Place Recognition Across Changing Appearance. In ICRA, 2020. 2


