Self-Supervised Visual Place Recognition Learning in Mobile Robots

Sudeep Pillai and John J. Leonard
CSAIL, MIT
{spillai, jleonard}@csail.mit.edu

Abstract— **Place recognition is a critical component in robot navigation that enables it to re-establish previously visited locations, and simultaneously use this information to correct the drift incurred in its dead-reckoned estimate. In this work, we develop a *self-supervised* approach to place recognition in robots. The task of visual loop-closure identification is cast as a metric learning problem, where the labels for positive and negative examples of loop-closures can be *bootstrapped* using a GPS-aided navigation solution that the robot already uses. By leveraging the synchronization between sensors, we show that we are able to learn an appropriate distance metric for arbitrary real-valued image descriptors (including state-of-the-art CNN models), that is specifically geared for visual place recognition in mobile robots. Furthermore, we show that the newly learned embedding can be particularly powerful in disambiguating visual scenes for the task of vision-based loop-closure identification in mobile robots.**

I. Introduction

Place recognition for mobile robots is a long-studied topic [23] due to the far-reaching impact it will have in enabling fully-autonomous systems in the near future. State-of-the-art methods for place-recognition today use hand-engineered image feature descriptors and matching techniques to implement their vision-based loop-closure mechanisms. While these model-based algorithms have enabled significant advances in mobile robot navigation, they are still limited in their ability to learn from new experiences and adapt accordingly. We envision robots to be able to learn from their previous experiences and continuously tune their internal model representations in order to achieve improved task-performance and model efficiency. With these considerations in mind, we introduce a bootstrapped mechanism to learn and fine-tune the model performance of vision-based loop-closure recognition systems in mobile robots.

With a growing set of experiences that a robot logs today, we recognize the need for *fully automatic solutions* for experience-based task learning and model refinement. Inspired by NetVLAD [1], we cast the problem of place recognition in mobile robots as a *self-supervised* metric learning problem. Most previous works [23, 28, 36] use hand-engineered image descriptors or pre-trained Convolutional Neural Network architectures [19, 39] to describe an image for classification or matching. All these methods, in some way or the other, require a hand-engineered *metric* for matching the visual descriptors extracted. The choice of feature extraction needs to be tightly coupled with the right distance metric in order to retrieve similar objects appropriately. This adds yet another level of complexity in designing and tuning reliable systems that are fault tolerant and robust to operating in varying appearance regimes. Furthermore, these approaches do not provide a mechanism to optimize for specific appearance regimes (e.g. learn to ignore fog/rain in those specific conditions). We envision that the distance metric for these feature descriptors be learned [3, 4] from experience, and that it should be done in a bootstrapped manner [20]. Furthermore, we would prefer that the features describing the same place to be repeatably embedded close to each other in some high-dimensional space, with the distances between them to be *well-calibrated*. With this self-supervised mechanism, we expect robots to be able to quickly adapt to the visual appearance regimes it typically sees, and reliably perform visual place recognition as it gathers more experience.

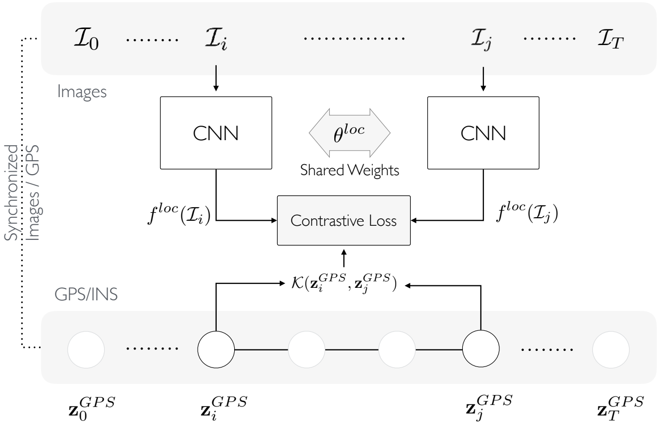


Fig. 1: **Self-Supervised Metric Learning for Localization** ▶ The illustration of our proposed self-supervised Siamese Net architecture. The model bootstraps synchronized cross-modal information (Images and GPS) in order to learn an appropriate similarity metric between pairs of images in an embedded space, that implicitly learns to predict the loop-closure detection task. The key idea is the ability to sample and train our model on positive and negative pairs of examples of similar and dissimilar places by taking advantage of corresponding GPS location information.

II. Related Work

Visual place recognition in the context of vision-based navigation is a well studied problem. In order to identify previously visited locations the system need to be able

Sudeep Pillai and John J. Leonard are with the Computer Science and Artificial Intelligence Lab (CSAIL), Massachusetts Institute of Technology (MIT), Cambridge MA 02139, USA. For more details, visit http://people.csail.mit.edu/spillai/learning-localization.

to extract salient cues from an image that describes the content contained within it. Typically, the same place may be significantly different from its previous appearance due to factors such as variation in lighting (e.g. sunny, cloudy, rainy etc), observed viewpoint (e.g. viewing from opposite directions, viewing from significantly different vantage points), or even perceptual aliasing (e.g. facing and seeing a brick-wall elsewhere). These properties make it challenging to hand-engineer solutions that robustly operate in a wide range of scenarios.

Local and Global methods Some of the earliest forms of visual place recognition entailed directly observing pixel intensities in the image and measuring their correlation. In order to be invariant to viewpoint changes, subsequent works [6, 7, 8, 18, 27, 34] proposed using low-level *local* and *invariant* feature descriptors. These descriptions are aggregated into a single high-dimensional feature vector via Bag-of-Visual-Words (BoVW) [31], VLAD [15] or Fisher Vectors [16]. Other works [28, 34, 35] directly modeled whole-image statistics and hand-engineered *global* descriptors such as GIST [30] to determine an appropriate feature representation for an image.

Sequence-based, Time-based or Context-based methods While image-level feature descriptions are convenient in matching, it becomes less reliable due to perceptual aliasing, or low saliency in images. These concerns led further work [9, 24, 26, 29] in matching whole sequences of consecutive images that effectively describes a place. In SeqSLAM, the authors [29] identify potential loop closures by casting it as a sequence alignment problem, while others [9] rely on temporal consistency checks across long image sequences in order to robustly propose loop closures. Mei et al. [27] finds cliques in the pose graph to extract place descriptions. Lynen et al. [24] proposed a placeless-place recognition scheme where they match features on the level of individual descriptors. By identifying high-density regions in the distance matrix computed from feature descriptions extracted across a large sequence of images, the system could propose swaths of potentially matching places.

Learning-based methods In one of the earliest works in learning-based methods Kuipers and Beeson [20] proposed a mechanism to identify distinctive features in a location relative to those in other nearby locations. In FABMAP [7], the authors approximate the joint probability distribution over the bag-of-visual-words data via the Chow-Liu tree decomposition to develop an information-theoretic measure for place-recognition. Through this model, one can sample from the conditional distribution of visual word occurrence, in order to appropriately weight the likelihood of having seen identical visual words before. This reduces the overall rate of false positives, thereby significantly increasing precision of the system. Another work from Latif et al. [22] re-cast place-recognition as a sparse convex L_1 minimization problem with efficient homotopy methods that enable robust loop-closure hypothesis. In similar light, experience-based learning methods [6, 8] take advantage of the robot's previous experiences to learn the set of features to match, incrementally adding more details to the description of a place if an existing description is insufficient to match a known place.

Deep Learning methods Recently, the advancements in Convolutional Neural Network (CNN) Architectures [19, 33] have drastically changed the landscape of algorithms used in vision-based recognition tasks. Their adoption in vision-based place recognition for robots [4, 36] have recently shown promising results. However, most domain-specific tasks require further model fine-tuning of these large-scale networks in order to perform reliably well. Despite the ready availability of training models and weights, we foresee the data collection and its supervision being a predominant source of friction for fine-tuning models for tasks such as place recognition in robots. Due to the rich amount of cross-modal information that robots typically collect, we expect to *self-supervise* tasks such as place recognition by fine-tuning existing CNN models with the experience they have accumulated. To this end, we fine-tune these feature representations specifically for the task of loop-closure recognition and show significant improvements in the precision-recall performance.

III. Background: Metric Learning

In this work we rely on metric learning to learn an appropriate metric for the task of place recognition in mobile robots. The problem of metric learning was first introduced as *Mahalanobis metric learning* in [38], and subsequently explored [21] with various dimensionality-reduction, information-theoretic and geometric lenses. More abstractly, metric learning seeks to learn a non-linear mapping $f(\cdot;\theta) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ that takes in input data pairs $(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathbb{R}^n$, where the Euclidean distance in the new target space $\|f(\boldsymbol{x}_i;\theta) - f(\boldsymbol{x}_j;\theta)\|_2$ is an approximate measure of *semantic* distance in the original space $\mathbb{R}^n$. Unlike in the supervised learning paradigm where the loss function is evaluated over individual samples, here, we consider the loss over pairs of samples $\mathcal{X} = \mathcal{X}_S \cup \mathcal{X}_D$. We define sets of similar and dissimilar paired examples $\mathcal{X}_S$, and $\mathcal{X}_D$ respectively as follows

$$\mathcal{X}_S := \{(\boldsymbol{x}_q, \boldsymbol{x}_s) \mid \boldsymbol{x}_q \text{ and } \boldsymbol{x}_s \text{ are in the } \textit{same} \text{ class}\} \quad (1)$$

$$\mathcal{X}_D := \{(\boldsymbol{x}_q, \boldsymbol{x}_d) \mid \boldsymbol{x}_q \text{ and } \boldsymbol{x}_d \text{ are in } \textit{different} \text{ classes}\} \quad (2)$$

and define an appropriate loss function that captures the aforementioned properties.

Contrastive Loss The contrastive loss introduced by Chopra et al. [5] optimizes the distances between positive pairs $(\boldsymbol{x}_q, \boldsymbol{x}_s)$ such that they are drawn closer to each other, while preserving the distances between negative pairs $(\boldsymbol{x}_q, \boldsymbol{x}_d)$ at or above a fixed margin α from each other. Intuitively, the overall loss is expressed as the sum of two terms with y being the indicator variable in identifying positive examples from negative ones,

$$\mathcal{L}(\theta) = \sum_{(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathcal{X}} y D_{ij}^2 + (1-y)\Big[\alpha - D_{ij}\Big]_+^2 \quad (3)$$

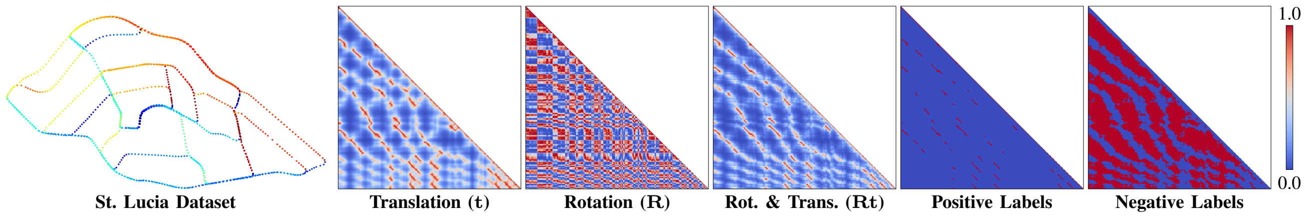


Fig. 2: **Bootstrapped learning using cross-modal information ▶** An illustration of the vehicle path traversed in the St. Lucia dataset (100909_1210) with synchronized Image and GPS measurements. The colors correspond to the vehicle bearing angle (Rotation $\mathbf{R}$) inferred from the sequential GPS measurements. The self-similarity matrix determined from the translation ($\mathbf{t}$), rotation ($\mathbf{R}$) and their combination ($\mathbf{Rt}$) on the St. Lucia Dataset using the assumed ground-truth GPS measurements. Each row and column in the self-similarity matrix corresponds to keyframes sampled from the dataset as described in Section IV-A. The sampling scheme ensures a time-invariant (aligned) representation where loop-closures appear as off-diagonal entries that are a fixed-offset from the current sequence (main-diagonal). We use a Gaussian kernel (Equation 6) to describe the similarity between keyframes and sample positive/negative samples from the combined $\mathbf{Rt}$ similarity matrix. The $\mathcal{K}$ kernel computed in Equation 6 is used to "supervise" the sampling procedure. **Positive Labels:** Samples whose kernel $\mathcal{K}(\mathbf{z}^{GPS}, \mathbf{z}'^{GPS})$ evaluates to higher than $\tau_p^{\mathbf{Rt}}$ are considered as positive samples (in red). **Negative Labels:** Samples whose kernel $\mathcal{K}(\mathbf{z}^{GPS}, \mathbf{z}'^{GPS})$ evaluates to lower than $\tau_n^{\mathbf{Rt}}$ are consider as negative examples (in red).

$$\text{where} \quad D_{ij} = \|f(\boldsymbol{x}_i;\theta) - f(\boldsymbol{x}_j;\theta)\|_2^2 \tag{4}$$

$$\text{and} \quad y = \begin{cases} 1 & \text{if } (\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathcal{X}_S, \\ 0 & \text{if } (\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathcal{X}_D \end{cases} \tag{5}$$

Training with Siamese Networks Learning is then typically performed with a Siamese architecture [2, 5], consisting of two parallel networks $f(\boldsymbol{x};\theta)$ that share weights θ amongst each other. The contrastive loss is then defined between the two parallel networks $f(\boldsymbol{x}_i;\theta)$ and $f(\boldsymbol{x}_j;\theta)$ given by Equation 3. The scalar output loss computed from batches of similar and dissimilar samples are then used to update the parameters of the Siamese network θ via Stochastic Gradient Descent (SGD). Typically, batches of positive and negative samples are provided in alternating fashion during training.

IV. Self-Supervised Metric Learning for Place Recognition

A. *Self-supervised dataset generation*

Multi-camera systems and navigation modules have more-or-less become ubiquitous in modern autonomous systems today. Typical systems log this sensory information in an asynchronous manner, providing a treasure of cross-modal information that can be readily used for transfer learning purposes. Here, we focus on the task of vision-based place recognition via a forward-looking camera, by leveraging synchronized information collected via standard navigation modules (GPS/IMU, INS etc.).

Sensor Synchronization In order to formalize the notation used in the following sections, we shall refer to $(\mathcal{I}_t, \mathbf{z}_t^{GPS})$ as the *synchronized* tuple of camera image $\mathcal{I}$, and GPS measurement $\mathbf{z}^{GPS}$, captured at approximately the same time t. In typical systems however, these sensor measurements are captured in an asynchronous manner, and the synchronization needs to be carried out carefully in order to ensure clean and reliable measurements for the bootstrapping procedure. It is important to note that for the specific task of place recognition, $\mathbf{z}$ can be also be sourced from external sensors such as inertial-navigation systems (INS), or even recovered from a GPS-aided SLAM solution.

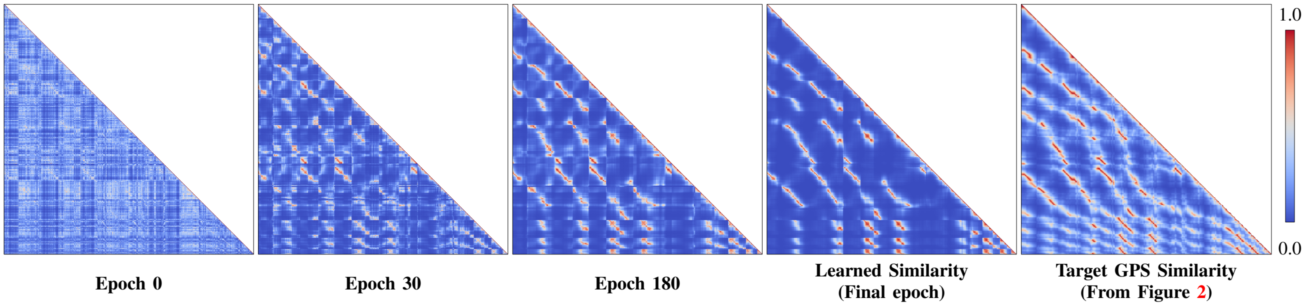
Keyframe Sub-sampling While we could consider the full set of synchronized image-GPS pairs, it may be sufficient to learn only from a diverse set of viewpoints. We expect that learning from this strictly smaller, yet diverse set, can substantially speed up the training process while being able to achieve the same performance characteristics when trained with the original dataset. While it is unclear what this sampling function may look like for image descriptions, we can easily provide this measure to determine a diverse set of GPS measurements. We incorporate this via a standard keyframe-selection strategy where the poses are sampled from a continuous stream whenever the relative pose has exceeded a certain translational or rotational threshold from its previously established keyframe. We set these translational and rotational thresholds to 5m, and $\frac{\pi}{6}$ radians respectively to allow for efficient sampling of diverse keyframes.

Keyframe Similarity The self-supervision is enabled by defining a viewing frustum that applies to both the navigation-view $\mathbf{z}_t$ and the image-view $\mathcal{I}$. We define a Gaussian similarity kernel $\mathcal{K}$ between two instances of GPS measurements $\mathbf{z}_i^{GPS}$ and $\mathbf{z}_j^{GPS}$ given by $\mathcal{K}(\mathbf{z}_i^{GPS}, \mathbf{z}_j^{GPS})$, (or $\mathcal{K}_{ij}$ in short):

$$\mathcal{K}_{ij} = \underbrace{\exp(-\gamma^{\mathbf{t}} \left\|\mathbf{z}_i^{\mathbf{t}} - \mathbf{z}_j^{\mathbf{t}}\right\|_2^2)}_{\text{Translation similarity}} \cdot \underbrace{\exp(-\gamma^{\mathbf{R}} \left\|\mathbf{z}_i^{\mathbf{R}} \ominus \mathbf{z}_j^{\mathbf{R}}\right\|_2^2)}_{\text{Rotation similarity}} \tag{6}$$

where $\mathbf{z}_i^{\mathbf{t}}$ is the GPS translation measured in metric-coordinates at time i, and $\mathbf{z}_i^{\mathbf{R}}$ is the corresponding rotation or bearing determined from the sequential GPS coordinates for the particular session (See Figure 2). Here, the only hyper-parameter required is the choice of the bandwidth parameters $\gamma^{\mathbf{R}}$ and $\gamma^{\mathbf{t}}$, and generally depends on the viewing frustum of the camera used. The resulting self-similarity matrix for the translation (using GPS translation $\mathbf{t}$ only), and the rotation (using established bearing $\mathbf{R}$ only) on a single session from the St. Lucia Dataset [13] is illustrated in Figure 2.

Distance-Weighted Sampling With keyframe based sampling considerably reducing the dataset for efficient training, we now focus on sampling positive and negative pairs in order to ensure speedy convergence of the objective function. We first consider the keyframe self-similarity matrix between all pairs of keyframes for a given dataset, and sample positive pairs whose similarity exceeds a specified threshold $\tau_p^{\mathbf{Rt}}$. Similarly, we sample negative pairs whose similarity



Epoch 0

Epoch 30

Epoch 180

Learned Similarity (Final epoch)

Target GPS Similarity (From Figure 2)

Fig. 3: **Self-Supervised learning of a visual-similarity metric ▶** An illustration of the similarity matrix at various stages of training. At *Epoch 0*, the distances between features extracted at identical locations are not well-calibrated requiring hand-tuned metrics for reliable matching. With more positive and negative training examples, the model at *Epoch 30* has learned to draw positive features closer together (strong red off-diagonal sequences indicating loop-closures), while pushing negative features farther apart (strong blue background). This trend continues with *Epoch 180* where the loop-closures start to look well-defined, while the background is consistently blue indicating a reduced likelihood for false-positives. Comparison of the learned visual-similarity metric against the target or ground truth similarity metric (obtained by determining overlapping frustums using GPS measurements). As expected, the distances in the learned model tend to be *well-calibrated* enabling strong precision-recall performance. Furthermore, the model can be qualitatively validated when the learned similarity matrix starts to closely resemble the target similarity matrix (comparing columns 2 and 3 in the figure).

is below $\tau_n^{\mathbf{Rt}}$. For each of the positive and negative sets, we further sample uniformly by their inverse distance in the original feature space following [37], to encourage faster convergence.

B. Learning an appropriate distance metric for localization

Our proposed self-supervised place recognition architecture is realized with a Siamese network with an appropriate contrastive loss [5] (given by Equation 7). This simultaneously finds a reduced dimensional metric space where the relative distances between features in the embedded space are *well-calibrated*. Here, well-calibrated refers to the property that negative samples are separated at least by a known margin α, while positive samples are likely to be separated by a distance less than the margin. Following the terminology in Section III, we consider tuples $(\mathcal{I}_i, \mathbf{z}_i^{GPS}) \in \mathcal{X}$ of similar (positive) $\mathcal{X}_S \subset \mathcal{X}$ and dissimilar (negative) examples $\mathcal{X}_D \subset \mathcal{X}$ for learning an appropriate embedding function $f^{loc}(\cdot;\theta^{loc})$. Intuitively, we seek to find a "*semantic measure*" of distance given by $D(\mathcal{I}_i,\mathcal{I}_j) = \left\| f^{loc}(\mathcal{I}_i;\theta^{loc}) - f^{loc}(\mathcal{I}_j;\theta^{loc}) \right\|_2$ in a target space of $\mathbb{R}^m$ such that they respect the kernel $\mathcal{K}_{ij}$ defined over the space of GPS measurements as given in Equation 6.

Let $(\mathcal{I}, \mathbf{z}^{GPS}) \in \mathcal{X}$ be the input data and $\mathbb{1}_G \in 0,1$ be the indicator variable representing dissimilar ($\mathbb{1}_G = 0$) and similar ($\mathbb{1}_G = 1$) pairs of examples within $\mathcal{X}$. We seek to find a function $f^{loc}(\cdot;\theta^{loc}) : \mathcal{I} \mapsto \Phi$ that maps the input image $\mathcal{I}$ to an embedding $\Phi \in \mathbb{R}^m$ whose distances between similar places are low, while the distances between dissimilar places are high. We take advantage of availability of synchronized Image-GPS measurements $(\mathcal{I}, \mathbf{z}^{GPS})$ to provide an indicator for place similarity, thereby rendering this procedure fully automatic or self-supervised. Re-writing equation 3 for our problem, we get Equation 7 where $D(\mathcal{I}_i,\mathcal{I}_j)$ measures the "*semantic distance*" between images (Equation 8).

$$\mathcal{L}(\theta^{loc}) = \sum_{\mathcal{X}} \mathbb{1}_G \cdot D_{ij}^2 + (1 - \mathbb{1}_G) \cdot \Big[\alpha - D_{ij}\Big]_+^2 \tag{7}$$

$$\text{where} \quad D_{ij} = \left\| f^{loc}(\mathcal{I}_i;\theta^{loc}) - f^{loc}(\mathcal{I}_j;\theta^{loc}) \right\|_2 \tag{8}$$

$$\text{and} \quad \mathbb{1}_G = \begin{cases} 1 & \text{if} \;\; \mathcal{K}(\mathbf{z}_i^{GPS}, \mathbf{z}_j^{GPS}) > \tau_p^{\mathbf{Rt}} \\ 0 & \text{if} \;\; \mathcal{K}(\mathbf{z}_i^{GPS}, \mathbf{z}_j^{GPS}) < \tau_n^{\mathbf{Rt}} \end{cases} \tag{9}$$

For brevity, we omit θ^{loc} and use $f^{loc}(\mathcal{I}_i)$ instead of the full expression $f^{loc}(\mathcal{I}_i;\theta^{loc})$. We pick the thresholds for $\tau^{\mathbf{Rt}}$ based on a combination of factors including convergence rate and overall accuracy of the final learned metric. Nominal values of $\tau_p^{\mathbf{Rt}}$ range from 0.8 to 0.9 that indicate the tightness of the overlap between viewing frustums of positive examples, with $\tau_n^{\mathbf{Rt}}$ for negative examples set to 0.4.

Figure 3 illustrates the visual self-similarity matrix of the feature embedding at various stages during the training process on the St. Lucia Dataset (100909_1210). At *Epoch 0*, when the feature embedding is equivalent to the original feature description, it is hard to disambiguate potential loop-closures due to the *uncalibrated* nature of the distances. As training progresses, the positively labeled examples of loop-closure image pairs are drawn closer together in the embedded space, while the negative examples are pushed farther from each other. As the training converges, we start to notice a few characteristics in the learned embedding that make it especially powerful in identifying loop-closures: (i) The red diagonal bands in the visual self-similarity matrix are well-separated from the blue background indicating that the learned embedding has identified a more separable function for the purposes of loop-closure recognition; and (ii) The visual self-similarity matrix starts to resemble the target self-similarity matrix computed using the GPS measurements (as shown in Figure 3). Furthermore, the t-SNE embedding[1] (colorization) of the learned features extracted at identical

[1] t-SNE [25] is a non-linear dimensionality reduction technique that is especially tailored to embedding high-dimensional data on a lower dimensional manifold, typically in $\mathbb{R}^2$ or $\mathbb{R}^3$. This makes it particularly valuable in visualizing high-dimensional data. In our case, we embed the high-dimensional features onto a 3-dimensional manifold via t-SNE and visualize the data as if they sit in a 3-dimensional RGB-colorspace. This allows us to identify similar feature embeddings by their color, where features with similar color indicate that they lie closer to each other in the original higher-dimensional space.

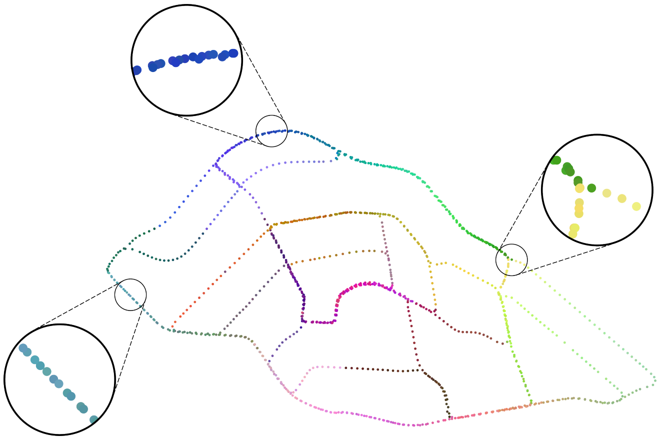


Fig. 4: **Trajectory with features embedded via T-SNE ▶** An illustration of the path traversed (100909_1210) with the colors indicating the 3-D t-SNE embedding of the learned features Φ extracted at those corresponding locations. The visual features extracted across multiple traversals along the same location are consistent, as indicated by their similar color embedding. colors are plotted in the RGB colorspace.

locations are strikingly similar, indicating that the learned feature embedding $f(\cdot;\theta^{loc})$ has implicitly learned a metric space that is more appropriate for the task of place-recognition in mobile robots.

C. Efficient scene indexing, retrieval and matching

One of the critical requirements for place-recognition is to ensure high recall in loop-closure proposals while maintaining sufficiently high precision in candidate matches. This however requires probabilistic interpretability of the matches proposed, with accurate measures of confidence in order to incorporate these measurements into the back-end pose graph optimization. Similarities or distances measured in the image descriptor space are not well-calibrated, making these measures only amenable to distance-agnostic matching such as k-nearest neighbor search. Moreover, an indexing and matching scheme such as k-nn also makes it difficult to filter out false positives as the distances between descriptors in the original embedding space is practically meaningless. Calibrating distances by learning a new embedding has the added advantage of avoiding these false positives, while being able to recover confidence measures for the matches retrieved.

Once feature embedding is learned, and the features Φ are mapped to an appropriate target space, we require a mechanism to insert and query these embedded descriptors from a database. We use a KD-Tree in order to incrementally insert features into a balanced tree data structure, thereby enabling $\mathcal{O}(\log N)$ queries.

V. Experiments and Results

We evaluate the performance of the proposed self-supervised localization method on the KITTI [11] and St. Lucia Dataset [13]. We compare our approach against the image descriptions obtained from extracting the activations from several layers in the Places365-AlexNet pre-trained model [41] (*conv3*, *conv4*, *conv5*, *pool5*, *fc6*, *fc7* and *fc8* layers). While we take advantage of the pre-trained models developed in [41] for the following experiments, the proposed framework could allow us to learn relevant task-specific embeddings from any real-valued image-based feature descriptor. The implementation details of our proposed method is described in detail in section V-C.

A. Learned feature embedding characteristics

While pre-trained models can be especially powerful image descriptors, they are typically trained on publicly-available datasets such as the ImageNet [32]. that have strikingly different natural image statistics. Moreover, some of these models are trained for the purpose of image or place classification. As with most pre-trained models, we expect some of the descriptive performance of Convolutional Neural Networks to generalize, especially in its lower-level layers (*conv1*, *conv2*, *conv3*). However, the descriptive capabilities in its mid-level and higher-level layers (*pool4*, *pool5*, *fc* layers) start to specialize to the specific data regime and recognition task it is trained for. This has been addressed quite extensively in the literature, arguing the need for domain adaptation and fine-tuning these models on more representative datasets to improve task-specific performance [10, 14].

Similar to previous domain adaptation works [10, 12, 14], we are interested in adapting existing models to newer task domains such as place-recognition with minimal human supervision involved. We argue for a self-supervised approach to model fine-tuning, and emphasize the need for a well-calibrated embedding space, where the features embedded in the new space can provide measures for both similarity and the corresponding confidence associated in matching.

Comparing performance between the original and learned embedding space In Figure 5, we compare the precision-recall performance in loop-closure recognition using the original and learned feature embedding space. For various thresholds of localization accuracy (20 and 30 meters), our learned embedding shows considerable performance boost over the pre-trained Places365-AlexNet model. In the figures, we also illustrate the noticeable drop in performance with the descriptive capabilities in the higher-level layers (*fc6*, *fc7*, *fc8*) as compared to the lower-level layers (*conv3*, *conv4*, *conv5*) in the Places365-AlexNet model. This is as expected, since the higher layers in the CNN (*pool5*, *fc6*, *fc7*) are more tailed to the original classification task they were trained for.

Embedding distance calibration As described earlier, our approach to learning an appropriate similarity metric for visual loop-closure recognition affords a probabilistic interpretation of the matches proposed. These accurate measures of confidence can be later used to incorporate these measurements into the back-end pose graph optimization. Figure 6 illustrates the interpretability of the proposed learned embedding distance compared to the original feature embedding distance. The histograms for the L_2 embedding distance separation is illustrated for both positive (in green) and negative (in blue) pairs of features. Here, a positive pair refers to feature descriptions of images taken at identical

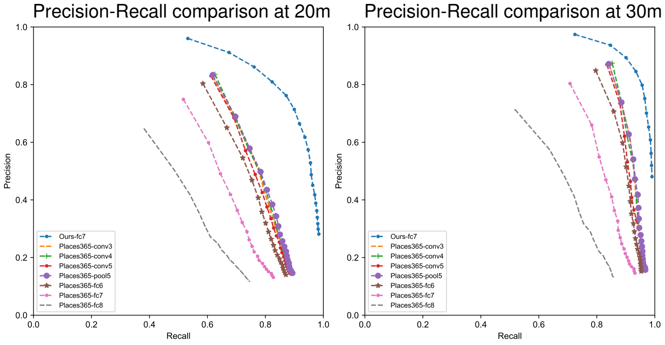


Fig. 5: **Precision-Recall performance in loop-closure recognition using the original and learned feature embedding space ▶** The figures show the precision-recall (P-R) performance in loop-closure recognition for various feature descriptors using the pre-trained Places365-AlexNet model and the learned embedding (*Ours-fc7*). Our learned embedding is able to significantly outperform the pre-trained Places365-AlexNet model for all feature layers, by *self-supervising* the model on a more representative dataset.

locations, while the negative pairs refer to pairs of feature descriptions that were taken from at least 50 meters apart from each other. The figure clearly illustrates how the learned embedding (*Ours-fc7*) is able to tease apart positive pairs, from those between the negative pairs of features, enabling an improved classifier (with a more obvious separator) for place-recognition. Intuitively, the histogram overlap between the positive and negative probability masses measures the ambiguity in loop-closure identification, with our learned feature embedding (*Ours-fc7*) demonstrating the least amount of overlap.

Nearest-Neighbor search in the learned feature embedding space Once the distances are calibrated in the feature embedding space, even a naïve fixed-radius nearest neighbor strategy, that we shall refer to as ε-NN, can be surprisingly powerful. In Figure 7, we show that our approach is able to achieve high-recall, with considerably strong precision performance for features that lie within distance α (contrastive loss margin as described in Section IV-B) from each other.

Furthermore, the feature embedding can also be used in the context of image retrieval with strong recall performance via naïve k-Nearest Neighbor (k-NN) search. Figure 8 compares the precision-recall performance of the k-NN strategy on the original and learned embedding space, and shows a considerable performance gain in the learned embedding space. Furthermore, the *recall* performance also tends to be higher for the learned embedding space as compared to the original descriptor space.

B. Localization performance within visual-SLAM front-ends

Figure 9 shows the trajectory of the optimized pose-graph leveraging the constraints proposed by our learned loop-closure proposal method. The visual place-recognition module determines constraints between temporally distant nodes in the pose-graph that are likely to be associated with the same physical location. To evaluate the localization module independently, we simulate drift in the odometry chains by injecting noise in the individual ground truth odometry measurements.

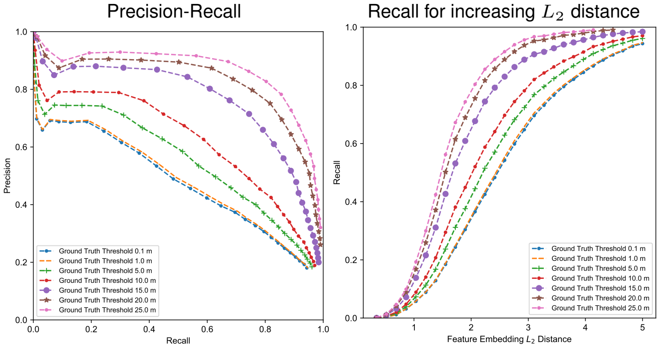


Fig. 7: **Precision-Recall (*Ours-fc7*) performance for loop-closure recognition in the original and learned feature embedding space using fixed-radius neighborhood search (ε-nn) ▶** The first column convincingly shows that our learned feature embedding space is able to maintain strong Precision-Recall performance by using ε-nn (fixed-radius search). The plot on the second column shows the recall performance with increasing feature embedding L_2 distance considered for each query sample. The Siamese network was trained with a contrastive loss margin of $\alpha = 10$, which distorts the embedding space such that positive pairs are encouraged to only be separated by an L_2 distance of 10 or lower. The figure on the *right* shows that in the learned feature embedding space (*Ours-fc7*), we are able to capture most candidate loop-closures within an L_2 distance of 5 from the query sample, as more matching neighbors are considered.

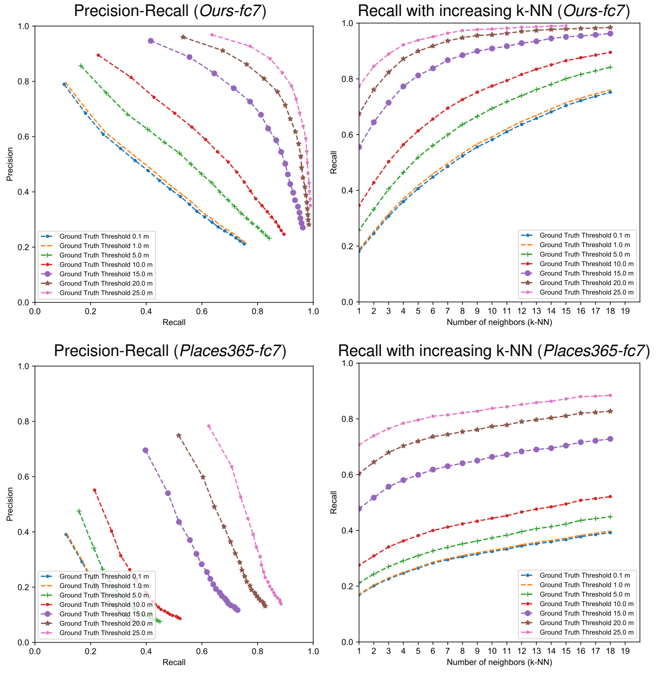


Fig. 8: **Precision-Recall performance for loop-closure recognition in the original and learned feature embedding space using k-Nearest Neighbors ▶** The first column shows that our learned feature embedding space is able to perform considerably better than the pre-trained layers (Places365-AlexNet *fc7*). The plot on the second column shows the recall performance with increasing set of neighbors considered for each query sample. Using the learned feature embedding space (Ours-fc7), we are able to capture more candidate loop-closures within the closest 20 neighbors of the query sample.

The trajectory recovered from sequential noisy odometry measurements are shown in red, as more measurements are added ($t_1 < t_2 < t_3 < T$). With every new image, the image is mapped into the appropriate embedding space and subsequently used to query the database for a potential loop-closure. The loop-closures are realized as weak zero rotation

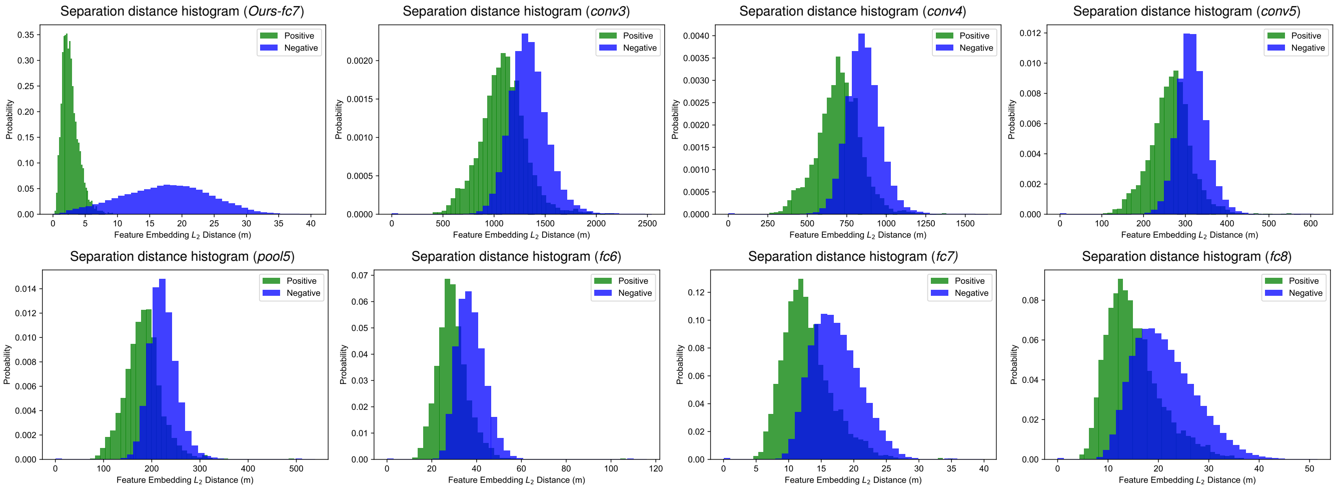


Fig. 6: **Separation distance calibration ►** The histograms of L_2 distances between positive and negative examples are shown for the various feature descriptions with the pre-trained Places365-AlexNet model. Our learned model is able to fine-tune intermediate layers and distort the feature embedding such that the distances between positive and negative examples (similar and dissimilar places) are well-calibrated. This is seen especially in the first plot (top row, far left *Ours-fc7*), where the probability mass for positive and negative examples are better separated with reduced overlap, while the other histograms are not well-separated in the feature embedding space.

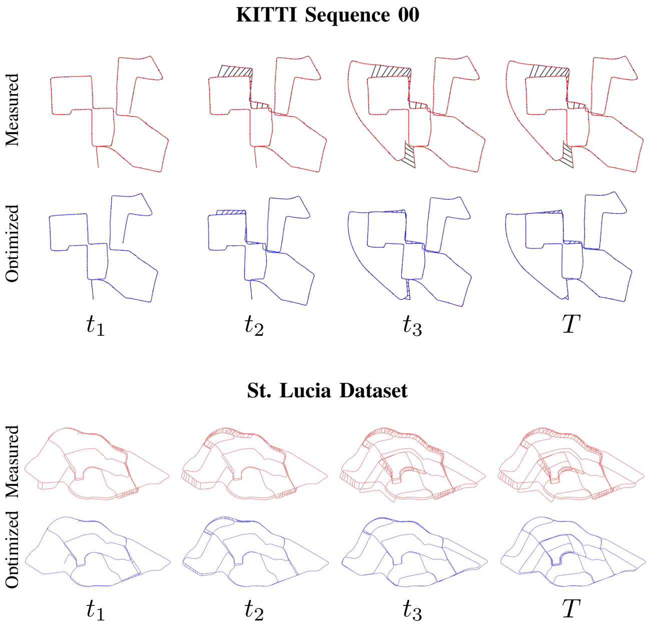


Fig. 9: **Vision-based Pose-Graph SLAM with our learned place-recognition module ►** The two sets of plots show the measured (in red) and optimized (in blue) pose-graph for a particular KITTI and St Lucia session. The crossed edges in the measured pose-graph corresponds to loop-closure candidates proposed by our learned place-recognition module. As more measurements are added and loop-closures are proposed ($t_1 < t_2 < t_3 < T$), the pose-graph optimization accurately recovers the true trajectory of the vehicle across the entire session. For both sessions, we inject odometry noise to simulate drift in typical odometry estimates.

and translation relative pose-constraints connecting the query node and the matched node. The recovered trajectories after the pose-graph optimization (in blue) shows consistent long-range, and drift-free trajectories that the vehicle traversed.

C. Implementation details

Network and Training We take the pre-trained Places205 AlexNet [40, 41], and set all the layers before and including *pool5* layer to be fixed, while the rest of the fully-connected layers (*fc6, fc7*) are allowed to be fine-tuned. The resulting network is used as a base network to construct the Siamese Network with shared weights (See Section IV-B). We follow the distance-weighted sampling scheme as proposed by Wu et al. [37], and sample 10 times more negative examples as positive examples. The class weights are scaled appropriately to avoid any class imbalance during training. In all our experiments, we set the sampling threshold $\tau^{\mathbf{Rt}}$ to 0.9, that ensures that identical places have considerable overlap in their viewing frustums. We train the model for 3000 epochs, with each epoch roughly taking 10s on an NVIDIA Titan X GPU. For most datasets including KITTI and St. Lucia Dataset, we train on 3-5 data sessions collected from the vehicle, and test on a completely new session.

Pose-Graph Construction and Optimization We use GTSAM[2] to construct the pose-graph and establish loop-closure constraints for pose-graph optimization. For validating the loop-closure recognition module, the odometry constraints are recovered from the ground truth, with noise injected to simulate dead-reckoned drift in the odometry estimate. They are incorporated as a relative-pose constraint parametrized in $SE(2)$ with $1\mathrm{e}{-3}$ rad rotational noise and $5\mathrm{e}{-2}$ m translation noise. We incorporate the loop-closure constraints as zero translation and rotation relative-pose constraint with a weak translation and rotation covariance of 3 m and 0.3 rad respectively. The constraints are incrementally added and solved using iSAM2 [17] as the measurements are recovered.

VI. Conclusion

In this work, we develop a *self-supervised* approach to place recognition in robots. By leveraging the synchronization between sensors, we propose a method to transfer and learn a metric for image-image similarity in an embedded space by sampling corresponding information from a

[2] http://collab.cc.gatech.edu/borg/gtsam

GPS-aided navigation solution. Through experiments, we show that the newly learned embedding can be particularly powerful for the task of visual place-recognition as the embedded distances are well-calibrated for efficient indexing and accurate retrieval. We believe that such techniques can be especially powerful as the robot can quickly fine-tune their pre-trained models to their operating environments, by simply collecting more relevant experiences.

References

[1] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5297–5307, 2016. 1

[2] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah. Signature verification using a" siamese" time delay neural network. In *Advances in Neural Information Processing Systems*, pages 737–744, 1994. 3

[3] Z. Chen, S. Lowry, A. Jacobson, Z. Ge, and M. Milford. Distance metric learning for feature-agnostic place recognition. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 2556–2563. IEEE, 2015. 1

[4] Z. Chen, A. Jacobson, N. Sunderhauf, B. Upcroft, L. Liu, C. Shen, I. Reid, and M. Milford. Deep learning features at scale for visual place recognition. *arXiv preprint arXiv:1701.05105*, 2017. 1, 2

[5] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005. 2, 3, 4

[6] W. Churchill and P. Newman. Practice makes perfect? managing and leveraging visual experiences for lifelong navigation. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 4525–4532. IEEE, 2012. 2

[7] M. Cummins and P. Newman. Appearance-only SLAM at large scale with FAB-MAP 2.0. *The International Journal of Robotics Research*, 30(9):1100–1123, 2011. 2

[8] P. Furgale and T. D. Barfoot. Visual teach and repeat for long-range rover autonomy. *Journal of Field Robotics*, 27(5):534–560, 2010. 2

[9] D. Galvez-Lopez and J. D. Tardos. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, 28 (5), October 2012. ISSN 1552-3098. doi: 10.1109/TRO.2012.2197158. 2

[10] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, pages 1180–1189, 2015. 5

[11] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012. 5

[12] X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 513–520, 2011. 5

[13] A. J. Glover, W. P. Maddern, M. J. Milford, and G. F. Wyeth. FAB-MAP + RatSLAM: Appearance-based SLAM for multiple times of day. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 3507–3512. IEEE, 2010. 3, 5

[14] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 999–1006. IEEE, 2011. 5

[15] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010. 2

[16] H. Jegou, F. Perronnin, M. Douze, J. Sánchez, P. Perez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE transactions on pattern analysis and machine intelligence*, 34(9): 1704–1716, 2012. 2

[17] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert. isam2: Incremental smoothing and mapping using the Bayes tree. *The International Journal of Robotics Research*, 31(2): 216–235, 2012. 7

[18] J. Košecká, F. Li, and X. Yang. Global localization and relative positioning based on scale-invariant keypoints. *Robotics and Autonomous Systems*, 52(1):27–38, 2005. 2

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1, 2

[20] B. Kuipers and P. Beeson. Bootstrap learning for place recognition. In *AAAI/IAAI*, pages 174–180, 2002. 1, 2

[21] B. Kulis et al. Metric learning: A survey. *Foundations and Trends® in Machine Learning*, 5(4):287–364, 2013. 2

[22] Y. Latif, C. Cadena, and J. Neira. Robust loop closing over time. *Robotics: Science and Systems VIII*, page 233, 2013. 2

[23] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford. Visual place recognition: A survey. *IEEE Transactions on Robotics*, 32(1):1–19, 2016. 1

[24] S. Lynen, M. Bosse, P. Furgale, and R. Siegwart. Placeless place-recognition. In *3D Vision (3DV), 2014 2nd International Conference on*, volume 1, pages 303–310. IEEE, 2014. 2

[25] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. 4

[26] W. Maddern, M. Milford, and G. Wyeth. CAT-SLAM: probabilistic localisation and mapping using a continuous appearance-based trajectory. *The International Journal of Robotics Research*, 31(4):429–451, 2012. 2

[27] C. Mei, G. Sibley, and P. Newman. Closing loops without places. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 3738–3744. IEEE, 2010. 2

[28] M. Milford. Vision-based place recognition: how low can you go? *The International Journal of Robotics Research*, 32(7):766–789, 2013. 1, 2

[29] M. J. Milford and G. F. Wyeth. SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1643–1649. IEEE, 2012. 2

[30] A. Oliva and A. Torralba. Building the gist of a scene: The role of global image features in recognition. *Progress in brain research*, 155: 23–36, 2006. 2

[31] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 2

[32] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 2015. 5

[33] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[34] N. Sünderhauf and P. Protzel. Brief-gist-closing the loop by simple means. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 1234–1241. IEEE, 2011. 2

[35] N. Sünderhauf, P. Neubert, and P. Protzel. Are we there yet? challenging SeqSLAM on a 3000 km journey across all four seasons. In *Proc. of Workshop on Long-Term Autonomy, IEEE International Conference on Robotics and Automation (ICRA)*, page 2013, 2013. 2

[36] N. Sunderhauf, S. Shirazi, A. Jacobson, F. Dayoub, E. Pepperell, B. Upcroft, and M. Milford. Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free. *Proceedings of Robotics: Science and Systems XII*, 2015. 1, 2

[37] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krhenbhl. Sampling Matters in Deep Embedding Learning, 2017. 4, 7

[38] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning with application to clustering with side-information. *Advances in neural information processing systems*, 15:505–512, 2003. 2

[39] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene CNNs. *arXiv preprint arXiv:1412.6856*, 2014. 1

[40] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014. 7

[41] B. Zhou, A. Khosla, A. Lapedriza, A. Torralba, and A. Oliva. Places: An image database for deep scene understanding. *arXiv preprint arXiv:1610.02055*, 2016. 5, 7