2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

# Resolving Place Recognition Inconsistencies Using Intra-Set Similarities

Peer Neubert, Stefan Schubert and Peter Protzel[1]

***Abstract*— Place recognition is the problem of finding associations between a query set of place descriptions and a database. It is an important means for loop closure detection in SLAM. The primary source of information to decide about associations is the pairwise similarity of descriptors between the query and the database items (e.g., image descriptor similarities). Beyond better descriptors, significant improvements were achieved by exploiting additional structural information, in particular by comparing sequences instead of individual items. In this paper, we propose to use another systematic source of information: intra-set similarities between items within the query or the database sets. They can be used to detect inconsistencies of groups of associations between database and query items, e.g. to inhibit matchings of multiple query descriptors to the same database descriptor if the query descriptors are mutually different. The underlying idea is a heuristic tightening of the triangle inequality of groups of descriptors. Based on a definition of matching inconsistencies, we propose an Inconsistency Resolution Procedure (IRP) to modify the inter-set similarities between database and query in a way that resolves existing inconsistencies with intra-set similarities. Our experiments show an average place recognition performance gain of >30 % in a general place recognition setup with 21 datasets and two state of the art image processing front-ends. The proposed approach does not require additional information beyond descriptor similarities, makes no assumptions of sequences, does not require training, and has no parameter that needs adjustment. It can be combined with other established techniques like descriptor standardization and sequence processing.**

## I. Introduction

Mobile robot place recognition is the problem of matching the current sensor information of a robot to a database of known places. A typical setup is visual place recognition where the robot's sensor is a camera and there is a set of images of known places. It is an important means for loop closure detection in simultaneous localization and mapping (SLAM) and for candidate selection for image based pose estimation. For example, some of the best performing localization approaches in the long-term visual localization challenge [1] combine a (holistic) place recognition approach to extract a small number of potential matching candidates from a large database of images of known places and use more elaborate feature-based methods for pose estimation. Place recognition is an intensively studied problem and many approaches have been proposed, including approaches that are able to recognize places despite severe changes of their appearance due to changing time of day, weather, or season.

This work was supported by the German Federal Ministry for Economic Affairs and Energy.

[1]All authors are with Faculty of Electrical Engineering and Automation Technology, Chemnitz University of Technology, Chemnitz, Germany {firstname.lastname}@etit.tu-chemnitz.de

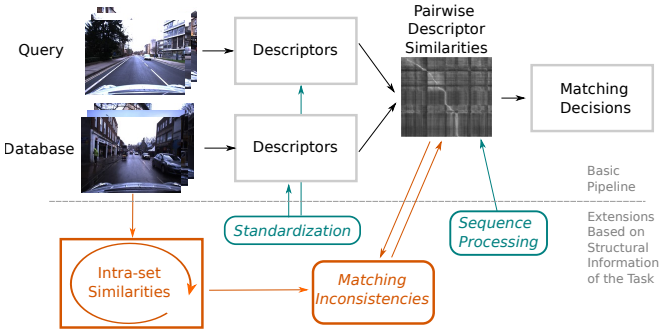


Fig. 1. The basic place recognition pipeline (above the horizontal dashed line) can be extended with additional information (below this line). Established approaches are standardization of descriptors and sequence processing. We propose to additionally use intra-set similarities to detect and resolve inconsistencies of groups of matchings. This can be implemented as a post-processing of the pairwise descriptor similarity matrix.

The upper part of Fig. 1 shows a basic processing pipeline for visual place recognition. In the basic problem setting, place recognition is a pure image retrieval problem with two sets of images: a database and a query set. Typically, the essential source of information is a pairwise distance estimate between database and query images based on some form of image descriptor comparison. When embedding image retrieval in the context of mobile robotics, even without using additional sensors of the robots, often, there is additional information that can be exploited. This is illustrated in the lower part of Fig. 1. For example, database and query image sets often have sequential structure which allows a combined evaluation of the distances of groups of temporally neighbored images - only match a query to a database image if the previous (and potentially also the subsequent images) from the query match to the corresponding database images. Exploiting such sequence information showed significant performance improvement in place recognition, particularly under changing changing environmental conditions [2]–[6]. Another structural property that can be exploited to address such changing environments is consistency and structure of changes, e.g., there might be only limited variation of environmental conditions *within* a sequence. This can be used by simple techniques like statistical standardization of descriptors [7], [8] or more sophisticated techniques like change removal [9].

In this paper, we exploit an additional systematic source of information as illustrated in the blocks highlighted in orange in Fig. 1. Descriptors for query and database items can not only be used to compute inter-set similarities between these two sets, but also to compute intra-set similarities within each set. Given these different types of similarities, we can find inconsistencies in groups of matchings between query and

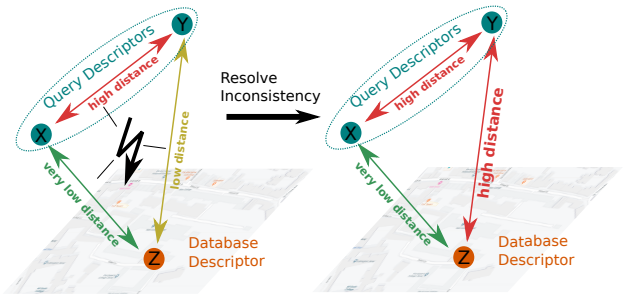


Fig. 2. The left part shows an example of an inconsistency: If two query images X and Y are mutually very different (i.e., they have a high descriptor distance), we consider it inconsistent, if both are associated to the same database image Z. Our simple heuristic approach to resolve the inconsistency is to increase the distance between Z and the less similar descriptor Y.

database items. Sec. III will provide a formal definition of matching inconsistencies and present an Inconsistency Resolution Procedure (IRP) to resolve them. The experimental evaluation in Sec. IV will demonstrate the practical benefit of IRP for place recognition. This algorithm is potentially useful for a wide application and integration in existing pipelines due to its simplicity, no requirements of additional sensor information, lack of parameters that need adjustment, and compatibility with existing techniques like sequence processing and descriptor standardization. Although all our experiments are performed using images, any source of distance information between dataset items can potentially be used. Code is available.[1]

## II. Related Work

Place recognition is a well studied problem, please refer to [10] for an overview. A variety of approaches can be used to create the pairwise descriptor matrix in Fig. 1. In particular in changing environments, descriptors based on deep convolutional neural networks are used [11]. Typical examples are descriptors from early convolutional layers of general purpose networks like AlexNet [12], specially trained holistic place recognition descriptors like NetVLAD [13] or according local features like DELF [14].

Our approach exploits information about distances between query and database images to inhibit certain matchings. In previous work [15], we exploited intra-database similarities to select candidate matchings to decrease the number of required image comparisons. In a related direction, pose information can be used to evaluate relations between items within the database or query sets. [16] uses GPS to constrain potential matches between database and query in order to reduce the number of image comparisons. [5] exploits "rough location priors (e.g., from a noisy GPS)" to deal with loops in database and query. [17] uses images with corresponding GPS-data to perform a self-supervised place descriptor learning. Experience maps [18], [19] group multiple representations of individual places. [20] learns offline condition-invariant broad-region detectors from beforehand collected images with a variety of appearances at particular locations.

[1] https://www.tu-chemnitz.de/etit/proaut/prstructure

Our matching inconsistency formulation is similar to a multi-label classification view on place recognition: all descriptors of the same place are considered as one class, matching descriptors without a common class is inconsistent. This is similar to sparse-coarse-coded function approximators (CMACs) like Coarse Coding and tile coding that are used in reinforcement learning [21]. [22] describes a LiDAR based classification approach for localization that uses a similar tiling of the spatial world. PlaNet [23] formulates geolocation of images as classification task; please refer to the survey [24] for related approaches. [25] uses a classification formulation to learn descriptors for place recognition (but not for inference).

Existing approaches that exploit structural information of the place recognition problem include change removal [9], simple descriptor normalization [7], more advanced standardization methods [8], as well as sequence processing approaches, e.g. [2]–[6]. In the experiments in Sec. IV, we will evaluate the combination of the proposed approach with descriptor standardization [8] and SeqSLAM [2]. In case of consistent conditions *within* the database and *within* the query set, a standardization by simply subtracting the mean descriptor from each each set showed to significantly improve the place recognition performance [8] (the same paper also proposes techniques to address changes within each set). The back-end of SeqSLAM [2] is a post-processing of the pairwise similarity matrix between database and query sets in order to find short linear sequences of neighbored high similarities. An requirement for application of SeqSLAM is that database and query sets are ordered sequences of images with constant velocity within each sequence and only small velocity deviations between sequences.

## III. Algorithmic Approach

This paper addresses place recognition setups with two sets, a database set $DB$ and a query set $Q$ where there might be environmental appearance changes between the two sets, but the conditions are more or less stable *within* each set. No additional structure (e.g. sequential ordering) or pose information (e.g. odometry) are required. The key idea is to not only use the inter-set similarities between query and the database entities $S^{DB \times Q}$, but also the intra-set similarities $S^{Q \times Q}$ and $S^{DB \times DB}$ between entities from the same set (i.e., $S^{Q \times Q}_{i,j}$ is the similarity of query entities $i$ and $j$). In practice, the similarities can e.g. be obtained from comparison of image descriptors.

### A. The inconsistency heuristic

Using the above inter- and intra-set similarities allows to identify inconsistencies in groups of matchings between query and database images as illustrated in Fig. 2. More formally, we define a matching inconsistency as follows:

**Def. Matching Inconsistency:** *Matches of two images X and Y to the same image Z are inconsistent, if the similarity* $S_{X,Y}$ *is smaller than* $min(S_{Z,X}, S_{Z,Y})$.

For example in case of images: If the two query images X,Y show the same place as the database image Z, we expect

the intra-dataset similarity $S^{Q\times Q}_{X,Y}$ of image descriptors for the images X and Y from the same set (i.e., the query image sequence) to be more similar than the image descriptors of this place obtained from different sets: $S^{DB\times Q}_{Z,X}$ and $S^{DB\times Q}_{Z,Y}$ (inter-set similarities). Otherwise we consider the matching $(Z, X)$ and $(Z, Y)$ inconsistent. This exploits the initial assumption from the beginning of this section that there are no significant appearance changes within the query set, thus the appearance of the place is more consistent within the query sequence than between query and database sets.

This is an heuristic approach. In practice, a place describes a potentially large area in an environment. Two diametrically opposed query camera *positions* of images of this place can easily be spatially more distant than each of them to the database image position - but, very importantly, there is an upper bound given by the triangle inequality. It is important to note that we can not assume a strict triangle inequality in an *appearance space* where we mix spatial position (i.e., places) and descriptor distances. For example, due to dynamics in the environment (e.g. moving objects), multiple descriptors of the same place can unsystematically become very different (and due to visual aliasing, descriptors of different places can be very similar). However, based on the assumption from the beginning of this section, we consider it a useful *heuristic* for place recognition to apply the (even tightened) analogy to the triangle inequality provided by the above definition of matching inconsistencies.

Despite the heuristic nature of the algorithmic approach presented in the following Sec. III-B, we consider it an inherent property of the place recognition problem that the intra-set similarities in $S^{Q\times Q}$ and $S^{DB\times DB}$ provide additional information that can be used to improve the place recognition results obtained from the inter-set similarities $S^{DB\times Q}$. The practical value of our particular heuristic approach will be demonstrated in the experiments in Sec. IV.

### B. IRP: Inconsistency Resolution Procedure

This section describes a procedure to resolve inconsistencies given inter-set similarities $S^{A\times B}$ and intra-set similarities $S^{A\times A}$. In the example from the previous section and also Fig. 2 the set A was a query set and B a database. Sec. III-D will extend this to a more general place recognition setup where intra-set similarities from both $A$ and $B$ are exploited.

The general idea of our approach is to resolves inconsistencies by modifying the inter-set similarities $S^{A\times B}$. It is important to notice that we modify the similarity values and not the underlying descriptors, thus the modified values in $S^{A\times B}$ do no longer perfectly correspond to descriptor similarities. However, this allows to use the modified similarity matrix in $S^{A\times B}$ directly as a replacement in the subsequent processing steps of standard place recognition pipelines. There are various ways how $S^{A\times B}$ can be modified in order to resolve all matching inconsistencies according to the previous definition in Sec. III-A - including a trivial solution by setting all values in $S^{A\times B}$ to zero. Of course, this is not the intended solution since we want the modified values in $S^{A\times B}$ to resemble the original descriptor similarities between sets $A$ and $B$ as close as possible.

**Algorithm 1:** Inconsistency Resolution Procedure (IRP)

**Data:** $S^{A\times B}$ the (sparse) similarity matrix of descriptor sets $A$ and $B$ $S^{A\times A}$ the (sparse) intra-set similarity matrix of descriptor set $A$
**Result:** An updated similarity matrix $S^{A\times B}$ without inconsistencies with respect to $S^{A\times A}$

1 **foreach** $b \in B$ **do**
  `// Create an empty clique of matchings to b`
2  $C_b = \emptyset$
  `// Sort elements in` $A$ `in order of decreasing similarities to` $b$ `stated in the according column from matrix` $S^{A\times B}$ `(called` $S^{A\times B}_b$`)`
3  $\hat{A} = sort(A, sort\_criterion = S^{A\times B}_b)$
  `// Process potential matchings in sorted order`
4  **foreach** $a \in \hat{A}$ **do**
    `// Add` $a$ `to the clique of matchings for` $b$
5  $C_b = C_b \cup a$
    `// Get the minimum intra-set similarity of elements from` $A$ `in the clique` $C_b$
6  $s_{min} = \min_{a_1 \in C_b, a_2 \in C_b} \mathbb{S}^{A\times A}_{a_1,a_2}$
    `// Update the similarity in` $S^{A\times B}$ `to the minimum of its previous value and the minimum intra-set similarity` $s_{min}$
7  $S^{A\times B}_{a,b} = min(S^{A\times B}_{a,b}, s_{min})$

Therefore, we propose the Inconsistency Resolution Procedure (IRP) listed in Algorithm 1. It is a simple greedy algorithm that processes matchings in order of decreasing similarity and only modifies a value in $S^{A\times B}$ if it is inconsistent with the set of more similar matchings. If this is the case, this value is modified by the least amount that resolves all inconsistencies with these more similar matchings. In particular, this leaves the best matchings unaltered, but uses their information together with intra-set similarities in $S^{A\times A}$ to improve other matchings.

Let us illustrate this algorithm with an example runthrough with $A$ being a set of query image descriptors and $B$ a set of known database image descriptors. We compute similarities $S^{A\times B}$ between query and database images, as well as similarities $S^{A\times A}$ between query images. Since the goal is to remove inconsistencies of matching multiple query images to the same database image, IRP can process each database image independently (line 1). An important data structure is the clique $C_b$ of all query images that were already matched to this database image $b$. All images in $C_b$ are supposed to show the same place in the world, thus we can expect them to have a high mutual similarity (that is why we call it a clique). The comparison of inter-set similarity values from $S^{A\times B}$ and the minimum intra-set similarity $s_{min}$ in this clique is used to identify matching inconsistencies. Starting from an empty clique $C_b$ (line 2), query images are processed in order of decreasing inter-set similarity to the current database image (lines 3 and 4).

This allows to resolve inconsistencies of new matchings with existing matchings by altering the inter-set similarity with the *smaller* value. Thus we exploit information of more similar inter-set matchings to modify less similar ones. The current query image $a$ is added to $C_b$ and the minimum intra-set similarity $s_{min}$ in the clique is updated (line 6). This value is used to modify the output similarity $S_{a,b}^{A\times B}$ between the current database image $b$ and the query image $a$ in line 7: If the clique-similarity $s_{min}$ is smaller than the original $S_{a,b}^{A\times B}$, then this matching is inconsistent with other, more similar matchings from the clique. By decreasing $S_{a,b}^{A\times B}$ to $s_{min}$, all inconsistencies of matching $a$ to $b$ with previous more similar query images are resolved.

After IRP finished, the resulting updated similarities $S^{A\times B}$ can then be used as replacement for the original similarities in the overall place recognition pipeline. The decision, whether a modified inter-set similarity leads to a matching can then be done, e.g., using a threshold (as in the precision-recall curve based evaluation in Sec. IV).

### C. Efficient implementation

This simple algorithm is easy to implement. Dependent on the size of the database and query sets, some implementation details can be crucial for an efficient implementation. Most importantly, it is not necessary to compute all values in $S^{A\times B}$ or $S^{A\times A}$, both can be sparse. For example, if there is a (approximate) k-nearest neighbor search used to find a small set of candidate database matchings for each query image, they can also be used to create a sparse matrix $S^{A\times B}$. However, this requires appropriate indexing methods to efficiently find all query images $a$ that have a database image $b$ in the set of nearest neighbors (in order to create $\hat{A}$ in line 3). The intra-set similarities in $S^{A\times A}$ can then be computed on demand.

Of course, the minimum clique-similarity $s_{min}$ from line 6 can be updated incrementally in each iteration of the inner loop from line 4 (since the clique only grows, the minimum similarity can be updated using the intra-set similarities of the new matching $a$ to all existing matchings in the clique).

Even if no (approximate) k-nearest neighbor search is used and $S^{A\times B}$ and $S^{A\times A}$ are dense matrices that are known in advance, we can use the same intuition to significantly speed up the computation. In most practical setups, we can preempt the inner loop (line 4) if we expect the database image to match only to a fraction of all query images. Runtime measures and an experimental evaluation of this preemption are presented in Sec. IV-C.

### D. Types of Place Recognition Problems and gIRP

Removing inconsistencies when matching multiple query images to the same database image does not address inconsistencies in the opposite direction: when different database images are matched to the same query image.

It is an interesting question whether matching multiple database images to a single query image is relevant at all. In a localization problem setup with a database of images with *known poses*, it is sufficient to correctly match a query image to a single database image - the relation to all other database images can be induced from their spatial poses.[2] In the following, we call this **Single Best Matching** setup. This type of place recognition problem can be addressed by the IRP approach as described in the previous section.

Although a comprehensive discussion of different types of place recognition problem setups is beyond the capabilities of this paper, the IRP approach can be easily extended to a more **General Place Recognition** setup like in [26] [15] [6] [8] that allows arbitrary many matchings between database and query images. This setup is practically relevant if we do not know the poses of the database images for sure. For example, when place recognition is used in a SLAM system for matching a query image to potentially *multiple* previously seen database images whose mutual relation is still under investigation. In particular, when the best matching has a high pose uncertainty, SLAM systems can significantly benefit from additional matchings.[3]

When allowing arbitrary many matchings of database and query images, we are interested in resolving inconsistencies in both directions. For inconsistencies when matching multiple query images to the same database image, we can use IRP as described in Sec. III-B, we will call this $IRP_Q$. To address inconsistencies in the opposite direction, when matching a query image to multiple database images, we can change the roles of query and database when calling Algorithm 1, i.e., $B$ becomes the query set and $A$ the database set, we call this $IRP_{DB}$.

Since the input and output of IRP are compatible, we can remove inconsistencies in *both* directions by using two consecutive calls, written as $IRP_{DB}(IRP_Q(S^{QvsDB}))$. However, since multiple calls of IRP are not commutative, the ordering of first removing inconsistencies with intra-database or intra-query similarities influences the results. To limit this influence, we propose to run both orderings and use the minimum resulting similarity, we refer to this more general IRP approach as **gIRP**:

$$gIRP = \min(IRP_{DB}(IRP_Q(S^{QvsDB})), \\ IRP_Q(IRP_{DB}(S^{QvsDB})))$$

Using gIRP increases runtime by about factor four (since IRP is called four times, however, two calls can be parallelized). The benefit from these multiple calls will be part of the experimental evaluation in the next section together with an evaluation of IRP in the single best matching problem setup.

[2] Although multiple matching *candidates* might be helpful to find a correct matching, the presented IRP approach does not help in this direction, since it might even be beneficial if the multiple matching candidates are inconsistent to increase their diversity.

[3] Similar to the Single Best Matching setup, IRP does not help to increase the diversity of candidates in the General Place Recognition setup. However, it can easily be used to find diverse *candidate groups* of consistent matchings by using different (diverse) initial candidate matchings in the first iteration of the inner loop in line 4 of Alg. 1.

## IV. Experimental Evaluation

### A. Experimental setup

We evaluate the presented IRP approach on a series of place recognition experiments. In principle, any approach that provides a pairwise similarity measure of observations of places can be used in combination with IRP. Here we use the two deep learning based visual front-ends NetVLAD [13] and AlexNet [12] to create the input similarity matrices $S$ (both for inter- and intra-set comparisons). For NetVLAD, we use the authors' version using VGG-16 and whitening trained on the Pitts30k dataset. For AlexNet, we use Matlab's ImageNet model and the output of the conv3 layer. We further use a Gaussian random projection to a 4,096 dimensional space for dimensionality reduction.

The evaluation is based on 21 comparisons from five datasets with different characteristics regarding environment, appearance changes, single or multiple visits of places, possible stops, or viewpoint changes. We use the same datasets as in [6]. **StLucia** (Various Times of the Day) [27]: Collected with a forward facing webcam mounted on a car driving in a suburb between morning and afternoon over several days. Each sequence contains several loop closures. We sampled images at 1Hz. **Oxford** RobotCar [28]: Recorded with a car equipped with several cameras and lidars over a period of over a year. The dataset is demanding due to seasonal changes, weather, long-term changes like roadworks and building construction, a few loop closures within each sequence and stops in front of traffic lights or intersections. We sampled images at 10Hz of the front facing part of the trinocular stereo camera. **CMU** Visual Localization [29]: Five car rides along a 8km route with possible stops, weather, and seasonal changes. There are no loop closures within a sequence. We use the left camera. **Nordland** [30]: Time and viewpoint synchronized rides along a single train track once in each season. We use the same image set as [26]. **Gardens Point** Walking [31]: Hand held camera on a single route on campus, two times at day and once at night with controlled viewpoint deviations. The dataset is special due to the outdoor/indoor location, many pedestrians, and severe lighting changes. Nordland and Gardens Point are time synchronized, for all other datasets, we used GPS for ground truth.

Input to the evaluation procedure is a pairwise similarity matrix, either obtained directly from cosine similarities of image descriptors or as the result of one or multiple post-processing steps (e.g. IRP). We run a series of thresholds on the similarities to obtain binary decisions about matchings. We use ground-truth information to count true-positives, false-positives, and false-negatives for each threshold. Using these numbers, we compute a point on the precision-recall curve for each threshold. For concise evaluation, we report average precision computed as the area under the precision-recall curve (using trapezoidal integration). In the Single Best Matching setup, for each query image, only the single database image with highest similarity is evaluated, in the General Place Recognition setup, multiple database matchings are allowed and required to achieve perfect recall. Thus, this second setup is considerably more difficult.

### B. Place recognition performance

Table I shows the results for NetVLAD and AlexNet front-ends on the 21 dataset combination, the bottom three rows summarize the achieved improvements of the IRP approach (and the other approaches). In the simpler Single Best Matching setup, for many dataset combinations, using the input similarity S directly already provides (very) high average precision values. For some dataset combinations, resolving matching inconsistencies using the proposed IRP approach increased the performance up to 28 %. The mean improvement is much lower (4.5 and 2.4 % improvement), but very importantly, the results never get worse.

In the more difficult General Place Recognition setup, the average improvement by IRP increases to about 18 %. Moreover, using the proposed generalized version gIRP further increases the average improvement to about 32 %. The best case improvement is significantly higher (220 %), however, in one case the gIRP heuristic caused a performance drop by 2 %.

The last four columns of Table I demonstrate the benefit of IRP in combination with two other techniques that exploit structural information: standardization and sequence processing. We implement standardization as subtraction of the mean descriptor value per set (cf. [8]) and sequence processing as a modified version of SeqSLAM [2] (without local contrast normalization and without a single matching constraint since the used place recognition setup required multiple matchings). It can be seen that the proposed approach can provide additional benefit to these existing techniques. However, the more techniques are combined and the higher the resulting already achieved performance, the smaller is the additional improvement.

If one were allowed to use only one of these additional techniques, then descriptor standardization is a simple to use and flexible approach that provides good results. However, the results in Table I show that IRP exploits a different type of systematic information and can be used together with standardization to further significantly improve the results. Since both approaches have only moderate and similar requirements we consider this a practically relevant combination. Sequence processing with the SeqSLAM core requries restrictions on the camera motion and is not always applicable. However, in particular the combination with the AlexNet front-end shows that there can be a considerable performance increase when combining all three approaches (here, the gain increased from 81% to 97% when adding IRP).

### C. Computational effort and approximation quality

Fig. 3 shows the runtime of our unoptimized Matlab implementation of IRP using a single core of an Intel(R) Core(TM) i7-7500U CPU@2.70GHz laptop. Runtimes are reported without the computation of image descriptor similarities since this strongly depends on the underlying descriptor

TABLE I

EXPERIMENTAL RESULTS OF **IRP** WITH INPUT SIMILARITY $\boldsymbol{S}$ USING NETVLAD OR ALEXNET DESCRIPTORS. METRIC IS AVERAGE PRECISION. BEST VALUES OF APPROACHES WITHOUT STANDARDIZATION OR SEQUENCE PROCESSING ARE BOLD. THE COLORED ARROWS INDICATE LARGE ($\geq$25% BETTER/WORSE) OR MEDIUM ($\geq$5%) DEVIATION COMPARED TO "INPUT $\boldsymbol{S}$" RESPECTIVELY TO THE VERSION WITHOUT IRP.

| | | | | Single Best Matching | | General Place Recognition | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Dataset | Database | Query | Input S | IRP | Input S | IRP | gIRP | Std | Std+gIRP | Std+Seq | Std+gIRP+Seq |
| NetVLAD front-end | Gardens Point Walking | day_left | night_right | 0.72 | 0.76 ↗ | 0.40 | 0.44 ↗ | **0.47** ↗ | 0.54 | 0.61 ↗ | 0.93 | 0.95 → |
| | Gardens Point Walking | day_right | day_left | 1.00 | 1.00 → | **0.97** | **0.98** → | **0.98** → | 0.98 | 0.99 → | 1.00 | 1.00 → |
| | Gardens Point Walking | day_right | night_right | 0.82 | 0.86 ↗ | 0.51 | 0.56 ↗ | **0.61** ↗ | 0.66 | 0.72 ↗ | 0.95 | 0.96 → |
| | Oxford | 2014-12-09-13-21-02 | 2014-12-16-09-14-09 | 0.98 | 0.99 → | 0.87 | 0.90 → | **0.93** ↗ | 0.92 | 0.95 → | 0.93 | 0.95 → |
| | Oxford | 2014-12-09-13-21-02 | 2015-02-03-08-45-10 | 0.97 | 0.98 → | 0.93 | 0.96 → | **0.97** → | 0.96 | 0.98 → | 0.96 | 0.97 → |
| | Oxford | 2014-12-09-13-21-02 | 2015-05-19-14-06-38 | 0.99 | 0.99 → | 0.83 | 0.93 ↗ | **0.97** ↗ | 0.91 | 0.98 ↗ | 0.95 | 0.99 → |
| | Oxford | 2015-05-19-14-06-38 | 2015-02-03-08-45-10 | 0.96 | 0.98 → | 0.85 | 0.92 ↗ | **0.95** ↗ | 0.94 | 0.97 → | 0.98 | 0.98 → |
| | StLucia | 100909_0845 | 190809_0845 | 0.84 | 0.89 ↗ | 0.41 | 0.51 ↑ | **0.57** ↑ | 0.57 | 0.62 ↗ | 0.84 | 0.87 → |
| | StLucia | 100909_1000 | 210809_1000 | 0.88 | 0.91 → | 0.47 | 0.56 ↗ | **0.62** ↑ | 0.60 | 0.66 ↗ | 0.87 | 0.89 → |
| | StLucia | 100909_1210 | 210809_1210 | 0.86 | 0.90 → | 0.51 | 0.60 ↗ | **0.63** ↗ | 0.63 | 0.67 ↗ | 0.88 | 0.88 → |
| | StLucia | 100909_1410 | 190809_1410 | 0.85 | 0.89 → | 0.38 | 0.47 ↗ | **0.54** ↑ | 0.57 | 0.59 → | 0.88 | 0.88 → |
| | StLucia | 110909_1545 | 180809_1545 | 0.76 | 0.83 ↗ | 0.27 | 0.40 ↑ | **0.48** ↑ | 0.43 | 0.54 ↑ | 0.75 | 0.85 ↗ |
| | Nordland | fall | spring | 0.83 | 0.86 → | 0.39 | 0.44 ↗ | **0.55** ↑ | 0.60 | 0.67 ↗ | 0.99 | 0.99 → |
| | Nordland | fall | winter | 0.30 | 0.37 ↗ | 0.06 | 0.07 ↑ | **0.10** ↑ | 0.24 | 0.31 ↑ | 0.96 | 0.92 → |
| | Nordland | winter | spring | 0.41 | 0.47 ↗ | 0.11 | 0.12 ↗ | **0.18** ↑ | 0.36 | 0.42 ↗ | 0.96 | 0.95 → |
| | Nordland | summer | spring | 0.79 | 0.82 → | 0.32 | 0.39 ↗ | **0.50** ↑ | 0.57 | 0.64 ↗ | 0.98 | 0.98 → |
| | Nordland | summer | fall | 0.94 | 0.97 → | 0.63 | 0.74 ↗ | **0.83** ↑ | 0.83 | 0.89 ↗ | 1.00 | 1.00 → |
| | CMU | 20110421 | 20100901 | 0.91 | 0.92 → | 0.73 | 0.76 → | **0.78** ↗ | 0.73 | 0.78 ↗ | 0.81 | 0.86 ↗ |
| | CMU | 20110421 | 20100915 | 0.93 | 0.94 → | 0.77 | **0.78** → | **0.79** → | 0.78 | 0.79 → | 0.85 | 0.87 → |
| | CMU | 20110421 | 20101221 | 0.88 | 0.89 → | 0.56 | **0.59** ↗ | 0.55 → | 0.59 | 0.57 → | 0.65 | 0.65 → |
| | CMU | 20110421 | 20110202 | 0.97 | 0.97 → | 0.61 | **0.70** ↗ | **0.71** ↗ | 0.71 | 0.75 ↗ | 0.82 | 0.87 ↗ |
| | Mean gain | | | (Reference) | 4.47 % | (Reference) | 14.55 % | 27.76 % | 49.65 % | 67.03 % | 178.06 % | 178.44 % |
| | Best gain | | | | 24.77 % | | 48.32 % | 76.54 % | 334.02 % | 454.77 % | 1632.19 % | 1562.23 % |
| | Worst gain | | | | 0.00 % | | 0.71 % | -2.22 % | 0.63 % | 1.72 % | 2.73 % | 2.82 % |

| | | | | Single Best Matching | | General Place Recognition | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Dataset | Database | Query | Input S | IRP | Input S | IRP | gIRP | Std | Std+gIRP | Std+Seq | Std+gIRP+Seq |
| AlexNet front-end | Gardens Point Walking | day_left | night_right | 0.32 | 0.37 ↗ | 0.09 | 0.10 ↗ | **0.14** ↑ | 0.18 | 0.24 ↑ | 0.47 | 0.58 ↗ |
| | Gardens Point Walking | day_right | day_left | 0.88 | 0.90 → | 0.56 | **0.61** ↗ | **0.62** ↗ | 0.53 | 0.63 ↗ | 0.87 | 0.92 ↗ |
| | Gardens Point Walking | day_right | night_right | 0.90 | 0.94 → | 0.49 | 0.56 ↗ | **0.61** ↑ | 0.65 | 0.74 ↗ | 0.90 | 0.96 ↗ |
| | Oxford | 2014-12-09-13-21-02 | 2014-12-16-09-14-09 | 0.95 | 0.96 → | 0.50 | 0.64 ↑ | **0.67** ↑ | 0.66 | 0.77 ↗ | 0.67 | 0.83 ↗ |
| | Oxford | 2014-12-09-13-21-02 | 2015-02-03-08-45-10 | 0.94 | 0.96 → | 0.61 | 0.71 ↗ | **0.73** ↗ | 0.85 | 0.85 → | 0.84 | 0.87 → |
| | Oxford | 2014-12-09-13-21-02 | 2015-05-19-14-06-38 | 0.94 | 0.96 → | 0.23 | 0.51 ↑ | **0.73** ↑ | 0.78 | 0.89 ↗ | 0.80 | 0.91 ↗ |
| | Oxford | 2015-05-19-14-06-38 | 2015-02-03-08-45-10 | 0.91 | 0.95 → | 0.34 | 0.63 ↑ | **0.82** ↑ | 0.89 | 0.94 ↗ | 0.93 | 0.96 → |
| | StLucia | 100909_0845 | 190809_0845 | 0.95 | 0.96 → | 0.58 | 0.65 ↗ | **0.66** ↗ | 0.65 | 0.67 → | 0.86 | 0.89 → |
| | StLucia | 100909_1000 | 210809_1000 | 0.93 | 0.94 → | 0.56 | 0.62 ↗ | **0.66** ↗ | 0.66 | 0.69 → | 0.90 | 0.92 → |
| | StLucia | 100909_1210 | 210809_1210 | 0.91 | 0.93 → | 0.53 | 0.60 ↗ | **0.63** ↗ | 0.66 | 0.68 → | 0.85 | 0.88 → |
| | StLucia | 100909_1410 | 190809_1410 | 0.96 | 0.97 → | 0.60 | 0.66 ↗ | **0.69** ↗ | 0.70 | 0.72 → | 0.92 | 0.94 → |
| | StLucia | 110909_1545 | 180809_1545 | 0.94 | 0.95 → | 0.59 | 0.65 ↗ | **0.66** ↗ | 0.67 | 0.70 → | 0.88 | 0.92 → |
| | Nordland | fall | spring | 0.99 | 0.99 → | 0.81 | 0.81 → | **0.85** → | 0.92 | 0.94 → | 1.00 | 1.00 → |
| | Nordland | fall | winter | 0.94 | 0.96 → | 0.62 | 0.67 ↗ | **0.70** ↗ | 0.80 | 0.86 ↗ | 1.00 | 1.00 → |
| | Nordland | winter | spring | 0.93 | 0.94 → | 0.59 | 0.65 ↗ | **0.71** ↗ | 0.84 | 0.89 → | 1.00 | 1.00 → |
| | Nordland | summer | spring | 0.99 | 0.99 → | 0.76 | 0.78 → | **0.83** ↗ | 0.89 | 0.93 → | 1.00 | 1.00 → |
| | Nordland | summer | fall | 1.00 | 1.00 → | 0.94 | **0.96** → | **0.96** → | 0.96 | 0.97 → | 1.00 | 1.00 → |
| | CMU | 20110421 | 20100901 | 0.87 | 0.89 → | 0.47 | 0.60 ↑ | **0.63** ↑ | 0.55 | 0.66 ↗ | 0.65 | 0.79 ↗ |
| | CMU | 20110421 | 20100915 | 0.90 | 0.90 → | 0.61 | **0.67** ↗ | 0.65 ↗ | 0.66 | 0.68 → | 0.76 | 0.81 ↗ |
| | CMU | 20110421 | 20101221 | 0.88 | 0.90 → | 0.38 | 0.44 ↗ | **0.51** ↑ | 0.44 | 0.44 → | 0.59 | 0.61 → |
| | CMU | 20110421 | 20110202 | 0.95 | 0.96 → | 0.34 | 0.40 ↗ | **0.48** ↑ | 0.43 | 0.52 ↗ | 0.52 | 0.68 ↑ |
| | Mean gain | | | (Reference) | 2.40 % | (Reference) | 20.75 % | 35.88 % | 41.35 % | 54.63 % | 80.76 % | 96.93 % |
| | Best gain | | | | 17.01 % | | 121.94 % | 219.69 % | 241.48 % | 290.11 % | 417.34 % | 532.96 % |
| | Worst gain | | | | 0.06 % | | -0.08 % | 2.69 % | -4.88 % | 3.90 % | 6.78 % | 6.78 % |

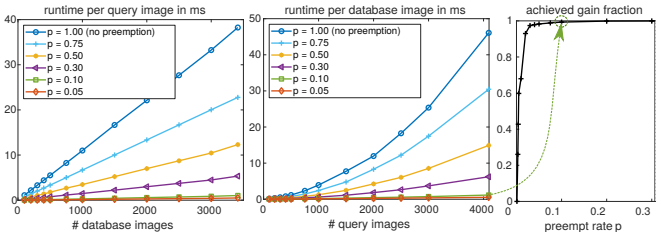


Fig. 3. *(left+mid)* We use the largest of all evaluated datasets Oxford 2014-12-09 vs. 2015-02-03 to measure the runtime for increasing numbers of database and query images for different preempt rates p. *(right)* The higher the preempt rate p, the higher is the performance gain. Here, we show the achieved fraction of the performance gain of IRP without preemption. The green arrow indicates that a value of $p = 0.1$ is sufficient for this dataset.

type (example below). We use the largest dataset from the previous evaluation (Oxford 2014-12-09 vs. 2015-02-03 with 3,413 and 4,094 images respectively). The left plot shows the runtime in milliseconds per query image for a varying number of database images (the number of query images is fixed to 4,094). The middle plot evaluates the runtime per database image for a varying query set size (the number of database images is fixed to 3,413). The runtime increases linearly in the number of database images but beyond linear in the number of query images. This is caused by the sorting in line 3 and the minimum similarity selection in the inner loop (line 6) in Alg. 1.

An approximation by preemptive abortion of this inner loop can significantly decrease the runtime. Without preemption, the runtime on this largest dataset is about 40 ms per query image, this can be reduced to 1 ms (4.2 seconds for the whole dataset) when using a preempt rate of $p = 0.1$. The preempt rate is the fraction of processed images from set A ($p \cdot |A|$ is the number of iterations of the inner loop in line 6 of Alg. 1). As can be seen on the right part of Fig. 3, using $p = 0.1$ is sufficient to achieve the full performance gain of using IRP on this dataset (the graph shows the ratio of the performance gain of IRP with $p < 1$ and IRP with $p = 1$).

In practice, the preempt rate should be chosen based on the expected maximum number of matchings for a single element from set B (cf. Alg. 1) and a reasonable choice can significantly vary between datasets. Fortunately, the larger the dataset, the smaller is the expected required preempt rate since then typically only a small fraction of all images will show the same place. A too low value of p is expected to reduce the performance gain by IRP but not cause a negative gain.

Our unoptimized Matlab implementation of IRP requires 4.2s for this largest dataset. Since gIRP involves four calls of IRP, its runtime is about 17s (about 4ms per query). This can be easily parallelized. Additionally, computing a similarity matrix $S \in \mathbb{R}^{3,413 \times 4,094}$ from the here used 4,096 dimensional image descriptors takes about 1.4s (for the whole dataset). For comparison, the runtime of the SeqSLAM core on this dataset and hardware can vary be-

tween 3:47min (OpenSeqSLAM [30]) and 2s (a considerably optimized custom version). The runtime for the descriptor standardization [8] is about 0.1s.

The memory requirements of IRP are low since no potentially large descriptors have to be stored, only indexes of images in cliques. The number of non-zero elements in the input similarity matrix is an upper bound for the total number of image indexes in all cliques (datastructure $C_b$ in Alg. 1).

## V. Conclusion

Place recognition pipelines can benefit from exploitation of additional structural knowledge of the underlying problem. A well-known example is to use sequential structure (e.g. in SeqSLAM). In this paper, we proposed to use a different systematic source of information: intra-set similarities within the query and database sets. The combination of these intra-set similarities together with the standard inter-set similarities (between query and database) allows to identify inconsistencies of matchings between database and query. We presented a simple Inconsistency Resolution Procedure (IRP) that alters the entries in an inter-set similarity matrix to resolve inconsistencies. We discussed efficient implementation and extended the IRP to a general place recognition setup where intra-set similarities from both query and database are used. The experimental evaluation demonstrated the practical value and average performance gains of about 32 % in the general place recognition setup. The presented IRP approach has no parameters that need adjustment (except for runtime optimization) and does not require training. It can be integrated in existing place recognition pipelines and is complementary to exiting techniques like sequence processing or descriptor standardization.

We consider intra-set similarities to be a valuable systematic source of information for this type of problem in general. The presented IRP is a simple and heuristic approach. Presumably, there are further, more sophisticated approaches that can more extensively exploit this type of information.

## References


[1] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, F. Kahl, and T. Pajdla, "Benchmarking 6dof outdoor visual localization in changing conditions," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[2] M. Milford and G. F. Wyeth, "Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights." in *Proc. of Int. Conf. on Robotics and Automation*, 2012.

[3] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera, "Towards life-long visual localization using an efficient matching of binary sequences from images," in *Proc. of Int. Conf. on Robotics and Automation*, 2015.

[4] P. Hansen and B. Browning, "Visual place recognition using hmm sequence matching," in *Int. Conf. on Intel. Robots and Systems*, 2014.

[5] O. Vysotska and C. Stachniss, "Lazy data association for image sequences matching under substantial appearance changes," in *IEEE Robotics and Automation Letters*, vol. 1, no. 1, 2016.

[6] P. Neubert, S. Schubert, and P. Protzel, "A neurologically inspired sequence processing model for mobile robot place recognition," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, 2019.

[7] S. Garg, N. Sünderhauf, and M. Milford, "Don't look back: Robustifying place categorization for viewpoint- and condition-invariant place recognition," in *Int. Conf. on Robotics and Automation (ICRA)*, 2018.

[8] S. Schubert, P. Neubert, and P. Protzel, "Unsupervised learning methods for visual place recognition in discretely and continuously changing environments," in *Int. Conf. on Rob. a. Autom. (ICRA)*, 2020.

[9] S. Lowry and M. J. Milford, "Supervised and unsupervised linear learning techniques for visual place recognition in changing environments," *IEEE Transactions on Robotics*, vol. 32, 2016.

[10] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *Trans. Rob.*, vol. 32, no. 1, 2016.

[11] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance of convnet features for place recognition," *Proc. of Int. Conf. on Intelligent Robots and Systems*, 2015.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012.

[13] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," *Trans. on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, 2018.

[14] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han, "Large-scale image retrieval with attentive deep local features," 10 2017, pp. 3476–3485.

[15] P. Neubert, S. Schubert, and P. Protzel, "Exploiting intra database similarities for selection of place recognition candidates in changing environments," *CVPR Workshop on Visual Place Recognition in Changing Environments*, 2015.

[16] O. Vysotska, T. Naseer, L. Spinello, W. Burgard, and C. Stachniss, "Efficient and effective matching of image sequences under substantial appearance changes exploiting gps priors," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[17] S. Pillai and J. Leonard, "Self-supervised visual place recognition learning in mobile robots," *IROS Workshop on Learning for Localization and Mapping*, 2017.

[18] W. Churchill and P. Newman, "Experience-based navigation for long-term localisation," *The International Journal of Robotics Research*, vol. 32, no. 14, pp. 1645–1661, 2013.

[19] C. Linegar, W. Churchill, and P. Newman, "Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 90–97.

[20] C. McManus, B. Upcroft, and P. Newmann, "Scene signatures: Localised and point-less features for localisation," in *Proceedings of Robotics: Science and Systems*, Berkeley, USA, July 2014.

[21] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2018.

[22] G. Kim, B. Park, and A. Kim, "1-day learning, 1-year localization: Long-term lidar localization using scan context image," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1948–1955, April 2019.

[23] T. Weyand, I. Kostrikov, and J. Philbin, "Planet - photo geolocation with convolutional neural networks," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 37–55.

[24] J. Brejcha and M. Cadik, "State-of-the-art in visual geo-localization," *Pattern Analysis and Applications*, 2017.

[25] Z. Chen, A. Jacobson, N. Sünderhauf, B. Upcroft, L. Liu, C. Shen, I. Reid, and M. Milford, "Deep learning features at scale for visual place recognition," in *Int. Conf. on Robotics and Automation*, 2017.

[26] P. Neubert and P. Protzel, "Beyond holistic descriptors, keypoints, and fixed patches: Multiscale superpixel grids for place recognition in changing environments," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, 2016.

[27] A. Glover, W. Maddern, M. Milford, and G. Wyeth, "FAB-MAP + RatSLAM: Appearance-based SLAM for Multiple Times of Day," in *Proc. of Int. Conf. on Robotics and Automation*, 2010.

[28] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 Year, 1000km: The Oxford RobotCar Dataset," *The Int. J. of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.

[29] H. Badino, D. Huber, and T. Kanade, "Visual topometric localization," in *Proc. of Intelligent Vehicles Symp.*, 2011.

[30] N. Sünderhauf, P. Neubert, and P. Protzel, "Are we there yet? challenging seqslam on a 3000 km journey across all four seasons," *Workshop on Long-Term Autonomy at Int. Conf. on Rob. a. Autom. (ICRA)*, 2013.

[31] A. Glover, "Day and night with lateral pose change datasets," 2014. [Online]. Available: https://wiki.qut.edu.au/display/cyphy/Day+and+Night+with+Lateral+Pose+Change+Datasets