Adaptive Thresholding for Sequence-Based Place Recognition

Olga Vysotska[1] Igor Bogoslavskyi[2] Marco Hutter[1] Cyrill Stachniss[3]

***Abstract*— Robots need to know where they are in the world to operate effectively without human support. One common first step for precise robot localization is visual place recognition. It is a challenging problem, especially when the output is required in an online fashion, and the current state-of-the-art approaches that tackle it usually require either large amounts of labeled training data or rely on parameters that need to be tuned manually, often per dataset. One such parameter often used for sequence-based place recognition is the image similarity threshold that allows to differentiate between pairs of images that represent the same place even in the presence of severe environmental and structural changes, and those that represent different places even if they share a similar appearance. Currently, selecting this threshold is a manual procedure and requires human expertise. We propose an automatic similarity threshold selection technique and integrate it into a complete sequence-based place recognition system. The experiments on a broad range of real-world and simulated data show that our approach is capable of matching image sequences under various illumination, viewpoint and underlying structural changes, runs online, and requires no manual parameter tuning while yielding performance comparable to a manual, dataset-specific parameter tuning. Thus, this paper substantially increases the ease of use of visual place recognition in real-world settings.**

I. Introduction

Any device moving through the real world, be it a mobile robot, autonomous vehicle, drone, or an augmented reality (AR) device, must be capable of recognizing places it knows from before as part of its ability to localize itself.

Recognizing places in the real world is challenging as environments typically do not remain static but undergo changes over time. These can be changes in illumination, appearance, geometry, or even a combination of any of the above. For example, an image taken during the daytime typically looks vastly different from one taken at night. In construction environments, a new wall might have been built between data acquisitions, changing both, the place's appearance and geometry. On a larger temporal scale, trees can change their colors and leaf coverage from summer to winter. All of these changes lead to tricky data association problems and make a robust solution to the problem of detecting previously seen places in real-world and large-scale environments a challenge.

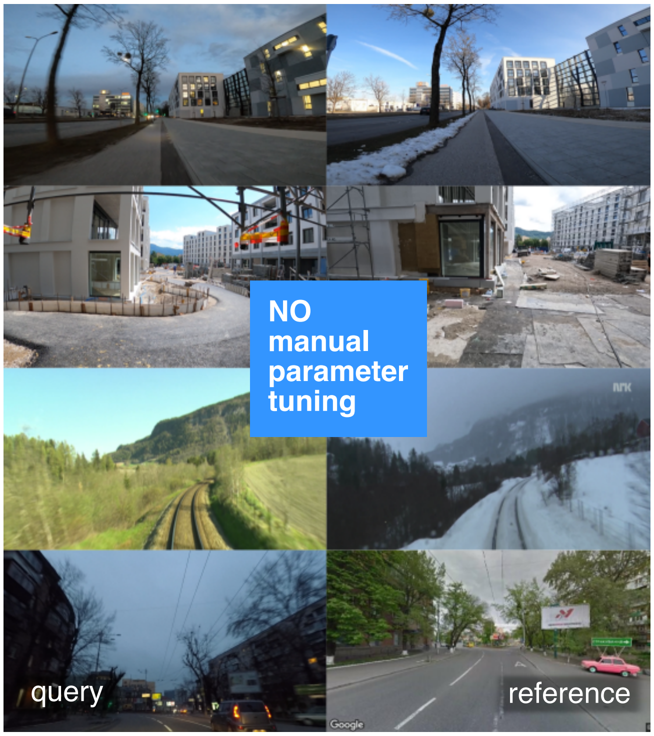

Fig. 1. Each row shows a pair of matching images from a different dataset. The query images are on the left and the reference ones are on the right. Our adaptive sequence-based place recognition approach was able to match all of these sequences without any manual parameter tuning.

Researchers in robotics and computer vision communities often aim to localize their devices using various sensors like LiDARs [2], [25], cameras [6], [20], [24], and even radars [4], [10], [17] in either maps or "previous experiences," often defined as sensor readings collected in the environment.

In this paper, we consider a monocular camera as our only sensor because cameras are cheap, small, and versatile. They can be found on most robots, AR devices, and even on phones. Such general availability provides the potential to develop a widely applicable framework capable of detecting previously visited places. Over the years, there has been steady progress in recognizing previously visited places from camera images by designing better traditional as well as deep-learning feature descriptors and treating place recognition as an end-to-end learning task. Some approaches use single images, while others consider image sequences or video streams to robustify the recognition of places. Despite all the progress made, there is still no perfect recognition approach and they all still can fail in the presence of drastic changes in the scene.

[1] ETH Zurich. [2] The Robotics and AI Institute, Switzerland. [3] Center for Robotics at the University of Bonn, Germany, the Lamarr Institute for Machine Learning and Artificial Intelligence, Germany, and with the Department of Engineering Science at the University of Oxford, UK.

This work has been funded by the Design++ Fellowship program of ETH Zurich as well as by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy, EXC-2070 – 390732324 – PhenoRob and under STA 1051/5-1 within the FOR 5351 (AID4Crops).

In this work, we utilize the fact that cameras operating in the real world, especially in robotics applications, generate *image streams*. We build upon our previous approach [22], which formulates sequence-based place recognition as a graph-based sequence matching technique that operates online and provides matching candidates between the current image stream (query) and existing reference sequences. Even though this approach has demonstrated strong performance even in complicated scenarios, its performance is sensitive to the so-called *similarity threshold*, i.e., a value that defines a minimum image similarity such that places are allowed to be considered the same place. Setting a good similarity threshold requires expertise and has often been done manually in the past [18]. This limitation, however, prevents the widespread use and deployment of such a system. The challenge of choosing a good threshold is exacerbated by the fact that robots require online operation in changing or novel environments, i.e., any manual parameter tuning should be avoided.

The main contribution of this paper is a sequence-based place recognition approach that uses, at its core, an adaptive technique to determine image similarity for sequence-based place recognition. Our approach adjusts the image similarity criterion based on the similarity values and matching results computed so far by learning a threshold over a small batch of data for every incoming query image. This approach adapts to changes *between image sequences*, e.g., the query sequence was recorded in summer while the reference one in winter, as well as the changes *within the sequences*, e.g., the images get progressively lighter or darker within a sequence, which is typical if the data were recorded at dusk or dawn outdoors, or when entering or leaving tunnels or buildings. Thus, our approach presented here turns our prior work [22] into an approach that requires no manual parameter tuning and is thus directly usable in new environments. Fig. 1 shows examples of various data on which our method is able to match sequences of images without any manual parameter tuning.

In sum, we present an image sequence-based place recognition approach that: (i) automatically selects the similarity threshold to provide consistent, reliable place recognition performance without the need for manual threshold selection; (ii) enables place recognition in long deployment scenarios with discrete and continuous changes within the query sequences; (iii) works in an online fashion and thus is suitable for mobile system deployment. The paper and our experimental evaluation back up all these claims.

II. Related Work

Visual place recognition is a well-studied topic, especially if seen as an image retrieval problem where for every image, we need to find an image that represents the same place in the database, if it exists.

Robots, however, perceive the world not in individual detached snapshots but rather as a stream or a sequence of images. Using such sequential data simplifies the problem of searching for similar images. Ho et al. [8] was one of the first to use sequence information to detect if the robot revisited a place it has visited before, also called loop closures. They construct and examine a similarity matrix and propose to use the Smith-Waterman algorithm to find significant local alignments similar to what we refer to as paths in a similarity matrix. Similar to this, Lynen et al. [13] propose a loop closure detection system that examines the similarity matrix and looks for the off-diagonal places with high local intensities for potential loop closures. In our approach, we are inspired by their use of the Kolmogorov-Smirnov statistical test to detect parts of the similarity matrix with potentially high-similarity regions. Cummins et al. [3] propose a loop closure detection system that has a probabilistic formulation and allows to search for loop closure candidates in the space of appearances rather than on the basis of individual images. Milford and Wyeth [15] propose to use a similar concept but in the context of alignment of two image sequences that were recorded in different period in time rather than for loop closure candidate selection in SLAM. They propose to fit lines to the similarity matrix and thus obtain correspondences between the query and the reference sequence. Naseer et al. [16] relax the assumption of linear movement and propose to match image sequences using graph-based formulation. In our previous work [22], [23], [24], we address the main limitations of their work and propose an approach for online place recognition where the matching hypothesis is updated on the fly for every incoming image.

The selection of the image feature descriptor is almost as important as the matching algorithm itself. Lowe [12] proposed a SIFT descriptor that is invariant to rotation, translation, and illumination changes. Even though the SIFT descriptor had an enormous impact and is still used today, its performance degrades in the presence of strong visual appearance changes, like seasonal changes, weather conditions, and day-night changes [21]. Recently proposed learning-based descriptors are able to tackle a variety of such challenges [1], [7], [11]. Our approach is orthogonal to the approaches that focus on finding better features for image matching and can use any such features. As better approaches emerge we directly benefit from their advances and are able to push our method even further. In this spirit, we rely on recently proposed DinoV2 SALAD features [9]. The majority of place recognition approaches need to estimate at some point if a pair of image or sequence descriptors [5], [14] represent the same place. This is typically done by manually setting some form of a similarity threshold. Neubert et al. [18] raises the point that setting such a threshold once is usually not enough to achieve good place recognition performance. They stress that in case of ”discrete” or ”continuous” changes within the query sequence the similarity threshold might change and needs adaptation to provide robust place recognition estimates. Motivated by these challenges, we tackle the problem of automatically selecting such a threshold for sequence-based place recognition approaches and present an adaptive similarity threshold selection approach that, when integrated into a sequence-based place recognition framework is able to

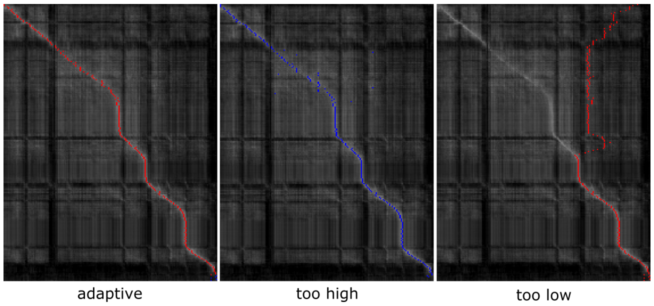

Fig. 2. Examples of various paths through a similarity matrix, showing the path correctly found by our adaptive method (left), as well as two situations when a similarity threshold was manually chosen to be either too high (middle) or too low (right) which leads to poor overall performance. When the similarity threshold is too high, all of the potential matches have low confidence, shown in blue. When it is too low, the matching process becomes over-confident and diverges from the real path.

match sequences of images that exhibit such ”discrete” and ”continuous” changes over a wide selection of real-world datasets without any manual parameter tuning.

III. Adaptive Sequence-Based Place Recognition

A. Sequence-based Place Recognition

We approach place recognition as an image sequence matching problem. Originally presented by Naseer et al. [16], the sequentiality of the data can be represented by a graph where every node n_{ij} represents the fact that a query image i was compared to a reference image j. The weights on the edges of this graph correspond to the similarity score of comparing two images. The most intuitive representation of such a graph is a similarity matrix, as seen in Fig. 2. Every element m_{ij} of this matrix stores the similarity score between a query image i and a reference image j. As a result, brighter parts of this matrix correspond to images with higher similarity, likely representing the same physical location in the environment. The task of matching image sequences then morphs into one of finding a path that follows brighter areas in a similarity matrix, such as shown by the red lines in Fig. 2 (left).

To be able to find such paths robustly, current state-of-the-art methods [22], [23] perform the search inside the graph structure and inevitably rely on a user-defined similarity threshold θ to differentiate between the nodes that represent the same place and those that do not. The nodes along the shortest path that are above θ, are *valid* nodes and are reported as image matches; the ones that are below θ are considered *hidden*. They are not considered image matches but are kept within the path to ensure sequentiality. It is normal to have a small number of hidden nodes along the path, which usually corresponds to a temporary obstruction. However, a higher number of hidden nodes usually indicates that the path hypothesis is wrong and must be revised. We refer the reader to our previous works [22], [23] for the details. In Fig. 2 and later in our experiments, we show valid nodes as red and hidden ones as blue.

The choice of the similarity threshold is critical for optimal performance of the sequence-based place recognition approach. Intuitively we want the path to follow bright lines in the similarity matrix. If the threshold is too high, only matches with very high similarity constitute valid matches and many nodes are considered hidden. If the threshold is too low, every match is considered valid and the path diverges easily. Fig. 2 shows all of these cases on a similarity matrix computed from real-world data.

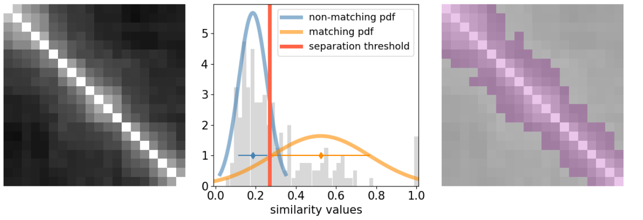

Fig. 3. For a patch of the similarity matrix (left), we fit a two-mode Gaussian mixture model to its values (middle) and, after obtaining the similarity threshold as the decision boundary between the Gaussian distributions, tint the similarity values above this threshold purple (right).

Note that selecting a good threshold is a complex task and requires expert knowledge as well as data observability. To address this, we propose an approach to set this similarity threshold automatically such that there is no need for an expert user to select it beforehand. Furthermore, by selecting this threshold automatically, our approach is able to adapt to changes that can occur *within* query sequence in long-term deployment scenarios.

B. Patch Selection Procedure

As a result of sequence-based place recognition, for every query image i, we have access to the current best path hypothesis. This hypothesis contains the matching estimate m_{ij} between the latest query i and some matching image j in the reference sequence, as well as for all the queries from 0 to i. We extract a patch over the similarity matrix of size p with its bottom right element corresponding to the latest best estimate m_{ij} and other elements being the similarities computed for the queries $i-p$. We continuously select these patches at every step and update the similarity threshold θ in a sliding window fashion.

C. Estimating Separation Threshold

If, for some query image i, we select a patch of values in the similarity matrix that were observed previously, as shown in Fig. 3, we observe that these values typically can be divided into those likely belonging to a valid path and those that do not. These two groups of similarity values form an unknown bimodal distribution, i.e., a distribution of valid matches and a distribution of non-valid ones. We seek to find a threshold that would reasonably separate these two distributions.

We approach this task, by fitting a 1D Gaussian mixture model (GMM) to our data as shown in the middle of Fig. 3. GMM provides us with means and standard deviations of two Gaussian distributions. Based on these parameters, we estimate the similarity threshold as the decision boundary between the two Gaussian distributions, a value θ such that:

$$P(\mathcal{N}(\mu_1, \sigma_1) \mid \theta) = P(\mathcal{N}(\mu_2, \sigma_2) \mid \theta), \qquad (1)$$

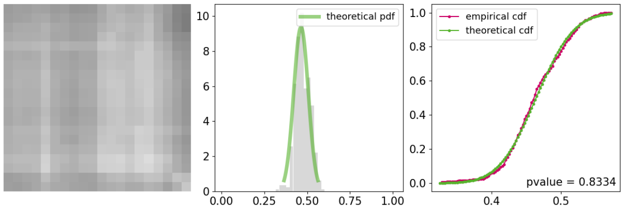

Fig. 4. A patch with no visible path (left). A unimodal Gaussian describes the underlying data well (middle). The KS test accepts the null hypothesis and reports that no path was detected in the patch.

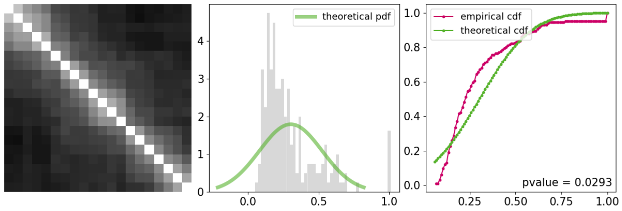

Fig. 5. A patch with a visible path (left). A unimodal Gaussian fits the similarity value distribution poorly (middle). The KS test rejects the null hypothesis because the distance between the CDFs is too big.

where $\mathcal{N}(\mu_1, \sigma_1)$ and $\mathcal{N}(\mu_2, \sigma_2)$ are the two Gaussian distributions estimated with the GMMs. Following the Bayes rule, using π_i to denote prior probabilities given by the GMM weights, and using $g(\theta, \mu_i, \sigma_i)$ to denote a probability density function of a Gaussian distribution, we can rewrite Eq. (1):

$$\pi_1 \; g(\theta, \mu_1, \sigma_1) = \pi_2 \; g(\theta, \mu_2, \sigma_2),$$

which, once expanded, becomes a quadratic equation:

$$\log\left(\frac{\pi_1 \sigma_2}{\pi_2 \sigma_1}\right) = \frac{(\theta - \mu_1)^2}{2\sigma_1^2} - \frac{(\theta - \mu_2)^2}{2\sigma_2^2}.$$

Solving this equation for θ we typically get one or two solutions, one of which lies between the means of our Gaussian distributions. If this solution exists, we choose it as the measurement of the current similarity threshold. We can use this threshold to decide which similarity values belong to a path and which do not. The right image in Fig. 3 shows values that belong to the path shaded purple.

To capture long-term trends in the similarity threshold rather than adapting to individual estimates, we smooth the similarity threshold update with a 1D Kalman filter (KF).

D. Statistical Test for Path Detection

The threshold estimation procedure described above relies on the presence of a valid path within the examined patch. Not every patch in the similarity matrix, however, contains a path. In fact most of randomly picked patches will not contain a discernible path. In typical sequence matching scenarios, even the best path hypothesis can lead to examining patches with no path at the start of the matching process, when the query and reference trajectories deviate or when the camera remains stationary for a prolonged time in both query and reference sequences.

Having no path in a patch means that the distribution of patch values is not bimodal, but rather unimodal. In this situation, running the GMM threshold estimation procedure, as described in Sec. III-C, can produce degenerate results. Moreover, the GMM can be a costly operation. To avoid unnecessary computation and unnecessarily updating the similarity threshold, we propose first checking if the patch contains a path. Inspired by Lynen et al. [13], we use the Kolmogorov-Smirnov statistical test (KS test) to check whether a patch under consideration contains a path, i.e. it is formed by a bimodal distribution.

Intuitively, the KS test allows one to test whether a set of given samples comes from a proposed distribution with known parameters. We assume that if the patch contains no discernible path as in Fig. 4 (left), the values are distributed according to a Gaussian distribution, while if there is a visible path as in Fig. 5 (left), the values are distributed differently. Formally, our objective is to reject a null hypothesis H_0 that states that the samples, that is, the values in the patch, originate from a Gaussian distribution with known parameters.

In order to test this hypothesis, we need to first find the parameters of the target Gaussian distribution, which we estimate by fitting a Gaussian distribution to all values found in the patch as seen in the middle images of Fig. 4 and Fig. 5. The KS test compares the cumulative density function (CDF) of this theoretical distribution with the empirical CDF computed directly on the patch values. If the difference between the theoretical and empirical CDFs is smaller than a ”critical value” defined by the KS test we accept the null hypothesis, see Fig. 4 (right). Otherwise, if the difference between CDFs is large, we reject the null hypothesis, see Fig. 5 (right). If the null hypothesis is accepted, we consider that there is no path found in the patch and do not use this patch for updating the similarity threshold.

Strictly speaking, rejecting the null hypothesis only gives a guarantee that the distribution of the values in the patch is not a unimodal Gaussian, but in our experience in most cases if this distribution is not unimodal the patch contains a path. Once we reject the null hypothesis, we proceed with the estimation of the similarity threshold as described in Sec. III-C and shown in Fig. 3. Note that the KS test provides statistical guarantees and in all our experiments we use the significance value of 0.05.

IV. Experimental Evaluation

The core of our place recognition system is an adaptive similarity threshold selection approach for sequence-based place recognition in changing environments and our experiments are designed to showcase its performance and to support our key claims, that (i) our adaptive threshold selection procedure delivers stable place recognition performance that does not depend on sensitive parameters and requires no expert user input, (ii) the adaptive nature of our approach allows for place recognition in the presence of discrete and continuous change within the data, and that (iii) our approach is applicable for online deployment scenarios where we might have only limited information about the similarity values within the patch used to learn the similarity threshold.

As a first experiment, we show that our approach is able to match images between sequences and provides stable place

TABLE I

F1 SCORES FOR VARYING INITIAL SIMILARITY THRESHOLDS

dataset	adaptive	previous		best one		best five	
		min	max	min	max	min	max
Bonn	0.91	0.0	0.99	0.0	0.99	0.01	1.0
Freiburg	0.88	0.0	0.98	0.0	0.98	0.0	1.0
Nordland	0.98	0.0	0.98	0.0	0.98	0.0	1.0
Construction	0.59	0.0	0.72	0.0	0.56	0.0	0.8
Discrete	0.99	0.67	0.99	0.26	0.99	0.26	1.0
Bike	0.91	0.0	0.9	0.0	0.99	0.0	0.98
Kyiv	0.88	0.0	0.96	0.0	0.96	0.0	0.97
max-min	-	0.83		0.88		0.9	

recognition results. For this, we ran our approach on a wide variety of data collected in the real world. These span from classical datasets like Nordland [19], or Freiburg and Bonn datasets [23], through our self-recorded datasets that consist of GoPro images exhibiting strong illumination changes that we match against the Google Street View imagery, to a dataset recorded in two sessions with a month between these sessions on a real construction site exhibiting both dynamic and structural changes in the environment. In all presented datasets, the images were extracted from recorded videos with 1 fps and 2 fps for handheld data collection in ”Construction”. The construction dataset is enabled by Design++ and ETH Zürich in collaboration with Halter AG. In the Nordland dataset, we used the first 500 images extracted with 1 fps, where winter represented the reference sequence and summer the query one, respectively. Moreover, the size of the patch is fixed to 20 images, as increasing the patch size to 30 or 50 does not lead to higher accuracy, according to our experiments.

We show the performance of our approach on all of these data in terms of F1 score in the *adaptive* column of the Tab. I. We compare our approach to the prior work in sequence-based matching [22] (*previous*) as well as two baseline strategies that select for every query the best candidate (*best one*) and five best candidates (*best five*). Here we operate on a fully precomputed similarity matrix as the *best one* and the *best five* strategies can only be used in this setting.

In Tab. I, we show that our approach (*adaptive*) exhibits similar place recognition performance to the *previous* sequence-based matching strategy in terms of an F1 score over a variety of datasets. In this experiment, we varied the initial similarity threshold for all the baseline methods by performing a full grid search for every dataset and show the best resulting place recognition performance in terms of F1 score in columns *max* and the minimal achieved score *min*. Please note that this choice is a complicated task and, in our experience, requires deep expert knowledge and often access to the ground truth information. The row *max-min* shows that the average difference between picking the best possible threshold and picking the worst one varies between 80 and 90% for the baseline methods, whereas the adaptive provides a potentially slightly worse but unique solution.

To showcase this further, we focus on the ”Construction” dataset from Tab. I. This is a very challenging dataset that contains strong structural changes between the query and the reference sequences. On the left side of Fig. 6, we show one such change where there is clutter, new wall panels, and a new door in the top image, which are not present in the reference sequence image shown at the bottom. These changes make picking the right threshold for matching images a hard task. The right side of Fig. 6 shows a precision-recall plot which we create by varying the similarity threshold. Through this plot, we observe that while it might be possible to manually pick a good similarity threshold, there are many more possibilities to pick one that leads to heavily degraded performance. Note that a choice of a good threshold is highly dataset-dependent and usually involves having access to all the pair-wise image matching scores between the query and reference sequences. Therefore, not only choosing a good threshold is hard but, in case of online operation, it might even be impossible, as such matching scores are not available beforehand. Our adaptive approach, produces a single result shown as a light-blue dot as there are no parameters tunable by the user. This result is only marginally worse (around 13%) than the result produced by picking the best possible parameters for the existing method. We believe that the absence of manual parameter tuning in our method is crucial to enable its stable real-world deployment and is one of its biggest strengths.

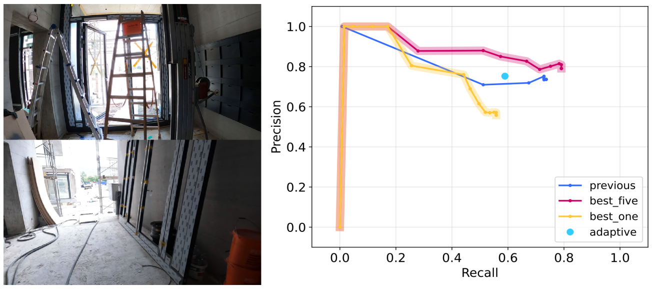

Fig. 6. The images (top: reference, bottom: query) are taken 1 month apart on an active construction site and reported as a match by our algorithm. The right side shows a precision-recall plot comparing the performance of various methods on this dataset with our approach labeled as ”adaptive”.

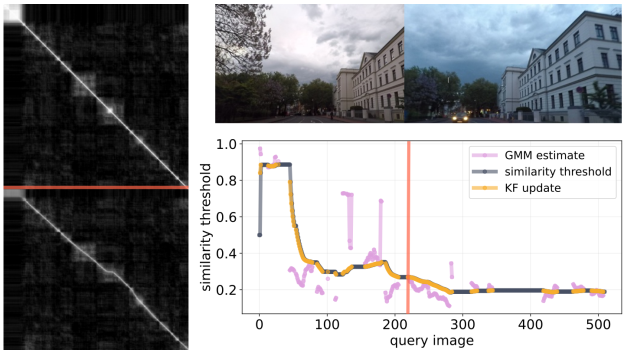

Fig. 7. Left: Similarity matrix where the top part exhibits day-to-day matching and the bottom part night-to-day matching. Bottom right: Similarity threshold evolution. The red line indicates the discrete change from day to night.

Our next two experiments are designed to support our second claim that our approach enables place recognition in long-term deployment scenarios with discrete and continuous changes within the query sequence.

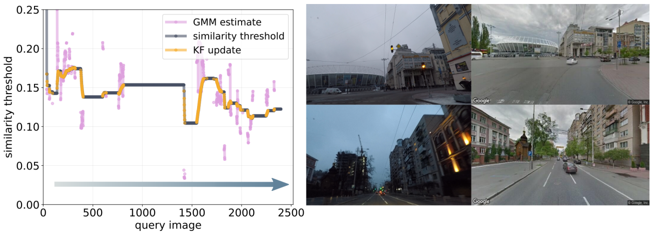

Fig. 8. Similarity threshold evolution in challenging scenario of matching GoPro camera imagery to Google street view images.

To showcase the adaptability of our approach to discrete changes within an image sequence, we look at two image sequences recorded from a car driving in Bonn. In both cases, the car drove the same route but once during the day and the other time during the night. We use the day dataset as our reference sequence and construct an artificial query sequence by stacking the day sequence on top of the night sequence. This approximates the scenario of a robot operating during the day and having near-perfect matching images in its reference sequence and then restarting its operation during the night, where the image matching scores become much worse. We illustrate how the similarity threshold learned by our method is able to cope with this situation in Fig. 7. We show the full similarity matrix on the left, a pair of representative images from day and night on the top-right and a plot that shows the evolution of the similarity threshold on the bottom-right. The two red lines, one across the similarity matrix and one across the similarity threshold evolution plot indicate the same spot in the query sequence where day abruptly changes to night. Comparing the adaptive similarity threshold values on both sides of the red line, we see that it converges to a new value in the presence of an abrupt discrete change in the query sequence. Overall, our method achieves the F1 score of 0.99 on this dataset, see the ”Discrete” label in Tab. I.

Similarly to discrete changes within an image sequence, our approach handles continuous changes during data acquisition. To showcase this, we collected a dataset in Kyiv, where the image sequence that we use as our query sequence was recorded at dusk with a GoPro camera mounted onto a car. As the sun sets, the images become progressively darker. We match this sequence against the Google Street View imagery collected during the day. This causes the images in the query and reference sequences to become progressively less visually similar. Fig. 8 shows illustrative examples of query-reference image pairs at the start (top image row) and towards the end of the route (bottom image row) as well as a plot of how our method adapts the similarity threshold continuously. Note how the similarity threshold is lower towards the end of the trajectory. Our approach successfully found most of the matching image pairs and reached the F1 score of 0.91 for this dataset denoted as ”Kyiv” in Tab. I.

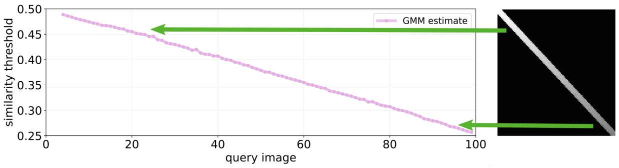

Fig. 9. When the similarity scores change over the course of the dataset, our approach is able to adapt the similarity threshold as shown here on simulated data.

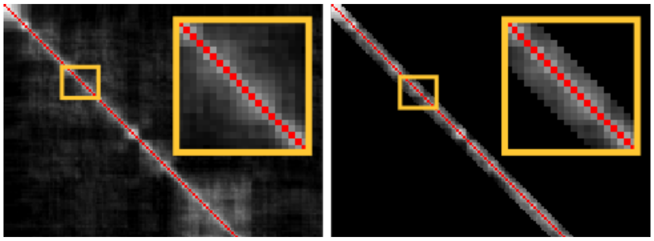

Fig. 10. (Left) Adaptive thresholding operates on the patch fully populated with similarity values (orange square). (Right) In online scenario, the patch for adaptive thresholding might not be complete. Our approach provides matches of equally high quality in both cases.

Furthermore, we perform an experiment in a controlled setting on simulated data. Here, we control the amount of change in the similarity values and observe the evolution of the similarity threshold estimated by our method. Fig. 9 shows a similarity matrix for perfectly aligned query and reference sequences where the simulated data becomes harder to match with time, making similarity values drop, which can be seen from a darkening diagonal. In our setup, the similarity of the non-matching images is very low, so the off-diagonal elements are dark. The corresponding similarity threshold evolution plot shows that our method behaves as we expect, and the threshold adapts to the observed similarities.

Our final experiment is designed to support our third claim that the performance of our adaptive similarity threshold selection does not degrade in an online setup. In this case, there is no pre-computed similarity matrix and we rely only on the similarity values computed during the graph search. Thus, it may happen that the patch with the similarity values is not complete. Fig. 10 shows that our place recognition approach works even in such setting and provides data associations of the same high quality as when it has access to all similarity values.

V. Conclusion

In this paper, we propose a sequence-based place recognition approach that eliminates the need for manual tuning of the image similarity threshold. Most approaches require a minimum image similarity threshold to consider image pairs a match – a parameter to be set by a human expert with access to ground truth, often in an offline fashion. We eliminate this manual tuning by proposing a way to estimate the parameter online during operation and without supervision, i.e., no positive or negative examples must be provided manually. We estimate the distribution of image similarities and determine a plausible separation by combining Gaussian mixture model estimation and statistical testing. Our results show that we perform similarly or only slightly worse than an approach operating with hand-tuned parameters.

References

[1] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[2] X. Chen, T. Läbe, A. Milioto, T. Röhling, O. Vysotska, A. Haag, J. Behley, and C. Stachniss. OverlapNet: Loop Closing for LiDAR-based SLAM. In *Proc. of Robotics: Science and Systems (RSS)*, 2020.

[3] M. Cummins and P. Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *Intl. Journal of Robotics Research (IJRR)*, 27(6):647–665, 2008.

[4] M. Gadd and P. Newman. Open-radvlad: Fast and robust radar place recognition, 2024.

[5] S. Garg and M. Milford. Seqnet: Learning descriptors for sequence-based hierarchical place recognition. *IEEE Robotics and Automation Letters*, 6(3):4305–4312, 2021.

[6] A. Glover, W. Maddern, M. Milford, and G. Wyeth. FAB-MAP + RatSLAM: Appearance-based slam for multiple times of day. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2010.

[7] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer. Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 14141–14152, 2021.

[8] K. Ho and P. Newman. Detecting Loop Closure with Scene Sequences. *Intl. Journal of Computer Vision (IJCV)*, 74:261–286, 2007.

[9] S. Izquierdo and J. Civera. Optimal transport aggregation for visual place recognition. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2024.

[10] H. Jang, M. Jung, and A. Kim. Raplace: Place recognition for imaging radar using radon transform and mutable threshold. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 10 2023.

[11] N. Keetha, A. Mishra, J. Karhade, K.M. Jatavallabhula, S. Scherer, M. Krishna, and S. Garg. Anyloc: Towards universal visual place recognition. *IEEE Robotics and Automation Letters (RA-L)*, 9(2):1286–1293, 2024.

[12] D. Lowe. Object recognition from local scale-invariant features. In *Proc. of the IEEE Intl. Conf. on Computer Vision (ICCV)*, 1999.

[13] S. Lynen, M. Bosse, P. Furgale, and R. Siegwart. Placeless place-recognition. In *Proc. of the Intl. Conf. on 3D Vision (3DV)*, volume 1, pages 303–310, 2014.

[14] R. Mereu, G. Trivigno, G. Berton, C. Masone, and B. Caputo. Learning sequential descriptors for sequence-based visual place recognition. *IEEE Robotics and Automation Letters (RA-L)*, 7(4):10383–10390, 2022.

[15] M. Milford and G. Wyeth. SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2012.

[16] T. Naseer, L. Spinello, W. Burgard, and C. Stachniss. Robust Visual Robot Localization Across Seasons using Network Flows. In *Proc. of the Conf. on Advancements of Artificial Intelligence (AAAI)*, 2014.

[17] G. Peng, H. Li, Y. Zhao, J. Zhang, Z. Wu, P. Zheng, and D. Wang. Transloc4d: Transformer-based 4d radar place recognition. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 17595–17605, June 2024.

[18] S. Schubert, P. Neubert, and P. Protzel. Unsupervised learning methods for visual place recognition in discretely and continuously changing environments. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, pages 4372–4378, 2020.

[19] 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.

[20] B. Talbot, S. Garg, and M. Milford. OpenSeqSLAM2.0: An Open Source Toolbox for Visual Place Recognition Under Changing Conditions. *arXiv preprint*, arXiv:1804.02156v2, 2018.

[21] C. Valgren and A. Lilienthal. SIFT, SURF & Seasons: Appearance-Based Long-Term Localization in Outdoor Environments. *Journal on Robotics and Autonomous Systems (RAS)*, 85(2):149–156, 2010.

[22] O. Vysotska and C. Stachniss. Lazy Data Association For Image Sequences Matching Under Substantial Appearance Changes. *IEEE Robotics and Automation Letters (RA-L)*, 1(1):213–220, 2016.

[23] O. Vysotska and C. Stachniss. Relocalization under substantial appearance changes using hashing. In *Proc. of the IROS Workshop on Planning, Perception and Navigation for Intelligent Vehicles*, 2017.

[24] O. Vysotska and C. Stachniss. Effective Visual Place Recognition Using Multi-Sequence Maps. *IEEE Robotics and Automation Letters (RA-L)*, 4(2):1730–1736, 2019.

[25] L. Wiesmann, T. Guadagnino, I. Vizzo, N. Zimmerman, Y. Pan, H. Kuang, J. Behley, and C. Stachniss. LocNDF: Neural Distance Field Mapping for Robot Localization. *IEEE Robotics and Automation Letters (RA-L)*, 8(8):4999–5006, 2023.