DEEP RECURRENT VISUAL PLACE RECOGNITION BASED ON SOFTMAX FINE-TUNED IMAGE REPRESENTATIONS

Jurica Maltar[1*] – Ivan Marković[2] – Domagoj Matijević[1] – Ivan Petrović[2]

[1]J. J. Strossmayer University of Osijek, School of Applied Mathematics and Informatics, Trg Lj. Gaja 6, Osijek, Croatia
[2]University of Zagreb, Faculty of Electrical Engineering and Computing, Unska 3, Zagreb, Croatia

ARTICLE INFO

***Article history*:**
Received: 29. 5. 2026.
Received in revised form: 5. 6. 2026.
Accepted: 6. 7. 2026.

Keywords:
Visual place recognition
Deep convolutional neural networks
Recurrent neural networks
Softmax regression
SeqSLAM

DOI: 10.30765/er.3383

Abstract:

Visual place recognition aims to identify previously visited locations from image representations that are robust to viewpoint and environmental changes. Deep convolutional neural network features perform well for this task, yet they are typically pretrained for classification or segmentation – objectives that do not account for the sequential nature of place recognition. In mobile robotics, exploiting image sequences can further improve performance. We propose softmax-based fine-tuning of a convolutional network extended with a recurrent model to enhance place recognition on image sequences. Experiments on two public datasets show that the proposed representation consistently outperforms competing approaches across all evaluated methods.

ACCEPTED

1 Introduction

This paper addresses visual place recognition (VPR) [1] – the problem of a vehicle, robot, or pedestrian visually recognizing a previously seen and labelled route. Formally, given a visual sensor, a query image $\boldsymbol{I_{q_i}}$ from a query database $\boldsymbol{Q}$ is compared against reference images $\boldsymbol{I_{d_j}}, \forall j \in \{1, \dots, |\mathcal{D}|\}$, from a reference database $\boldsymbol{\mathcal{D}}$. The comparison uses a similarity measure such as cosine similarity [2], [3], or a dissimilarity measure such as $\boldsymbol{\ell_2}$-distance [4] or sum of absolute differences [5]. VPR is challenging because the same route is traversed under varying environmental and non-environmental conditions. Figure 1 illustrates seasonal appearance changes on a Norwegian train route; similar issues arise with time-of-day variation, as in the Bonn dataset [2]. Viewpoint variance is also a concern: minimal for train-recorded data, modest for car-recorded data, and severe for pedestrian-recorded datasets such as Gardens Point [3]. Robust VPR therefore requires image representations that are both viewpoint- and condition-invariant.



Figure 1. The Nordland dataset used in visual place recognition experiments [1], [5]

Image representations for VPR have evolved from raw appearance [5], [6] and hand-crafted descriptors such as SIFT [7] and VLAD-style aggregations [8], to deep features extracted from convolutional neural networks. Depth images have also been explored to improve condition invariance [9]. Sünderhauf et al. [3] showed that AlexNet [10] convolutional features perform well, and that fine-tuning on Places205 [11] improves results further. Other backbones such as GoogLeNet [12] have also been used as feature extractors [13], [14]. NetVLAD [4] introduced an end-to-end differentiable VLAD layer trained with a triplet loss. HybridNet [15] was trained with softmax regression on large-scale surveillance imagery. More recent CNN-based approaches have substantially advanced large-scale and appearance-invariant VPR: CosPlace [16] and

* Corresponding author
E-mail address: jmaltar@mathos.hr

EigenPlaces [17] reformulate training for geo-localization at scale, while MixVPR [18], AnyLoc [19], and SALAD [20] focus on viewpoint- and domain-robust aggregations. Sequence-based deep models such as HVPR [21] fuse NetVLAD descriptors across image sequences.

Recurrent neural networks (RNNs) [22], including LSTM [23] and GRU [24] variants (see [25], [26] for surveys), capture temporal context and are therefore a natural fit for sequence-based VPR. Chancán and Milford [27] proposed an LSTM approach fusing visual and positional information. More recent sequence-aware methods continue to build on this direction [28]. Sequence-based place matching originated from Bayesian approaches such as FAB-MAP [29]. SeqSLAM [5] introduced matching over linear sequences and outperformed FAB-MAP, while cone-based SeqSLAM [30] and NOSeqSLAM [31] generalized the approach further.

In this paper, we propose softmax regression for VPR on sequentially captured images, adapting the image representation so that images of the same place become more similar while images of different places become more dissimilar. The contributions are as follows. First, a training pipeline that fine-tunes a DCNN extended with an RNN by forward-propagating sequences of feature descriptors extracted from vehicle-traversal imagery. Unlike existing DCNN+RNN approaches [27], which dismissed sequence-based matching algorithms as "complex hand-crafted heuristics" with questionable performance, our pipeline is designed to produce representations that are compatible with such algorithms. Second, an evaluation pipeline that couples the fine-tuned DCNN representation with the sequence-based algorithm NOSeqSLAM [31], demonstrating that the combination consistently outperforms competitive approaches – including HybridNet [15] and the original (non-fine-tuned) ResNet50 – in AUC on the Oxford RobotCar [32] and Nordland [33] datasets. Third, we show that softmax fine-tuning consistently improves performance across all evaluated place matching algorithms, suggesting that the benefit of fine-tuning is independent of the matching strategy.

ACCEPTED

2 Proposed approach to visual place recognition

Figure 2 illustrates the three stages of the proposed pipeline. Preprocessed images from $\boldsymbol{\mathcal{Q}}_{\text{train}}$ and $\boldsymbol{\mathcal{D}}_{\text{train}}$, collected sequentially during vehicle traversals, are used to fine-tune a DCNN augmented with an RNN via softmax regression (Section 2.1). The convolutional component of the fine-tuned DCNN then extracts feature maps from $\boldsymbol{\mathcal{Q}}_{\text{test}}$ and $\boldsymbol{\mathcal{D}}_{\text{test}}$ (Section 2.2), which are employed in sequence-based place matching (Section 2.3).

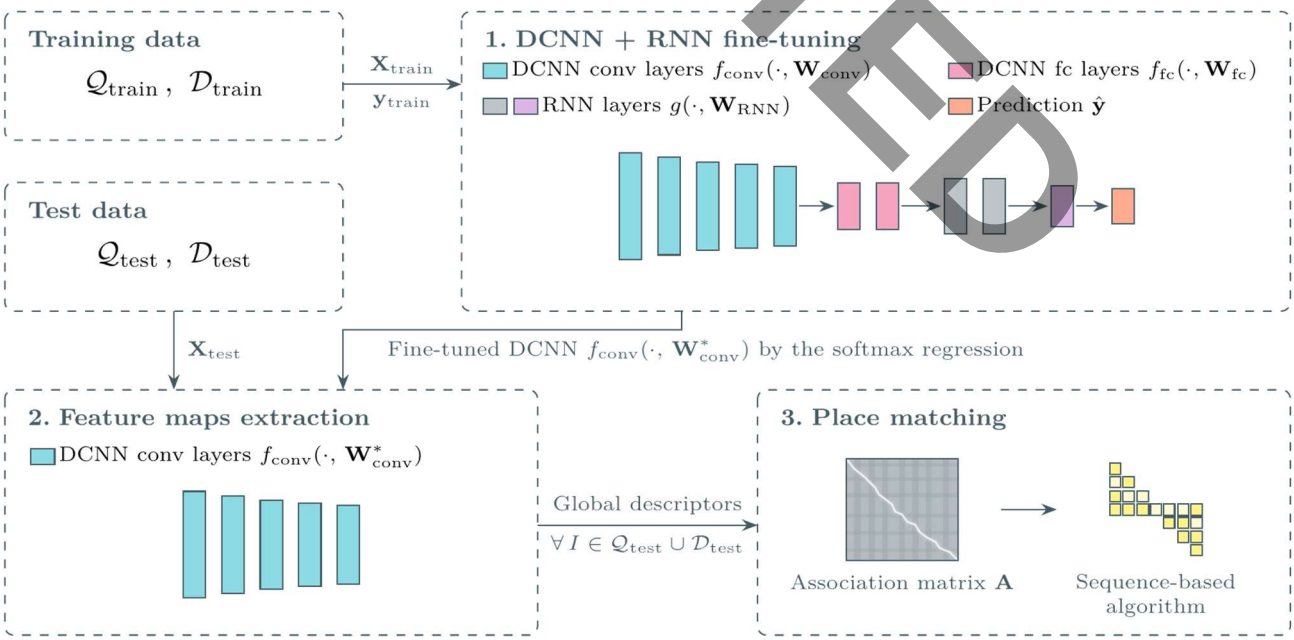


Figure 2. Overview of the proposed approach: (1) DCNN+RNN fine-tuning, (2) feature map extraction using the fine-tuned convolutional layers, and (3) sequence-based place matching

2.1 DCNN+RNN optimization via softmax regression

In multinomial classification, the objective is to assign an input vector $\boldsymbol{x}_\mathrm{k} \in \mathbb{R}^\mathrm{n}$ to one of $|\mathcal{C}|$ classes, where $\mathcal{C} = \{1, \dots, |\mathcal{C}|\}$ denotes the set of class labels. Posterior probability $p(\boldsymbol{y}_\mathrm{k} = c \mid \boldsymbol{x}_\mathrm{k};\ \boldsymbol{\Theta})$ is typically modeled as the composition of a linear mapping $\boldsymbol{\Theta} \in \mathbb{R}^{|\mathcal{C}|\times n}$ and the softmax function

$$\boldsymbol{\sigma}(\boldsymbol{x})[c] = \frac{\exp(\boldsymbol{x}[c])}{\sum_{\mathrm{j}\in\mathcal{C}} \exp(\boldsymbol{x}[j])},\ \forall c \in \mathcal{C}. \tag{1}$$

Given a set of independent and identically distributed samples $\boldsymbol{X} = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_\mathrm{m}]^\mathrm{T} \in \mathbb{R}^{m\times n}$ with corresponding labels $\boldsymbol{y} = [\boldsymbol{y}_1, \dots, \boldsymbol{y}_\mathrm{m}]^\mathrm{T} \in \mathbb{R}^{m\times|\mathcal{C}|}$, the model parameters are obtained by maximizing the log-likelihood

$$\boldsymbol{\Theta}^* = \mathit{argmax}_{\boldsymbol{\Theta}} \sum_{k=1}^{m} \sum_{\mathrm{c}\in\mathcal{C}} \boldsymbol{y}_\mathrm{k}[c] \log \boldsymbol{\sigma}(\boldsymbol{\Theta}\boldsymbol{x}_\mathrm{k})[c]\,. \tag{2}$$

This formulation is commonly referred to as softmax regression. In the special case $|\mathcal{C}| = 2$, it reduces to logistic regression. Since linear mappings are insufficient to model the nonlinear manifolds on which complex visual data typically lie, the linear term $\boldsymbol{\Theta} \cdot \boldsymbol{x}_\mathrm{k}$ can be replaced by a nonlinear mapping $f\colon \mathbb{R}^\mathrm{n} \to \mathbb{R}^{|\mathcal{C}|}$, such as a deep convolutional neural network (DCNN).

To apply this framework to sequential visual place recognition, let $\boldsymbol{I}_{\boldsymbol{q}_i} \in \boldsymbol{\mathcal{Q}}$ and $\boldsymbol{I}_{\boldsymbol{d}_j} \in \boldsymbol{\mathcal{D}}$ denote query and reference RGB images, respectively. For each reference image $\boldsymbol{I}_{\boldsymbol{q}_i}$, the ground-truth correspondence set $\boldsymbol{\mathcal{GT}}(\boldsymbol{I}_{\boldsymbol{d}_j})$ contains all query images matching that location. The label $\boldsymbol{y}_\mathrm{k}$ for an image $\boldsymbol{I}_\mathrm{k}$ is defined as

$$\boldsymbol{y}_k = \begin{cases} \left[\mathbb{1}_{I_k\in\boldsymbol{\mathcal{GT}}(\boldsymbol{I}_{\boldsymbol{d}_1})}\ \mathbb{1}_{I_k\in\boldsymbol{\mathcal{GT}}(\boldsymbol{I}_{\boldsymbol{d}_2})}\ \dots\ \mathbb{1}_{I_k\in\boldsymbol{\mathcal{GT}}\left(\boldsymbol{I}_{\boldsymbol{d}_{|\mathcal{D}|}}\right)}\right], & \boldsymbol{I}_k \in \boldsymbol{\mathcal{Q}} \\ \boldsymbol{e}_k, & \boldsymbol{I}_k \in \boldsymbol{\mathcal{D}} \end{cases} \tag{3}$$

where $\boldsymbol{e}_k$ is the k-th one-hot vector from the $|\boldsymbol{\mathcal{D}}|$-dimensional canonical basis.

For a fixed image $\boldsymbol{I}_\mathrm{k}$, a temporal window of length L is considered, consisting of $\lfloor L/2 \rfloor$ preceding and $\lfloor L/2 \rfloor$ succeeding frames. Each image in the sequence is processed by the DCNN $f(\cdot,\ \boldsymbol{W}_\mathrm{DCNN})$ to produce a sequence of feature maps, which are then passed through a recurrent neural network $g(\cdot,\ \boldsymbol{W}_\mathrm{RNN})$. The resulting outputs are concatenated and mapped by a linear projection $\boldsymbol{W}_\mathrm{LP}$ to produce the final prediction vector $\widehat{\boldsymbol{y}}_k$.

Training is performed by maximizing the log-likelihood with respect to all network parameters $\boldsymbol{W} = (\boldsymbol{W}_\mathrm{DCNN}, \boldsymbol{W}_\mathrm{RNN}, \boldsymbol{W}_\mathrm{LP})$,

$$\max_{\boldsymbol{W}} \sum_{k=1}^{m} \sum_{c\in\mathcal{C}} \boldsymbol{y}_\mathrm{k}[c] \log \widehat{\boldsymbol{y}}_k[c] \tag{4}$$

where the prediction $\widehat{\boldsymbol{y}}_k$ is obtained by applying the softmax function to the output of the linear projection over the concatenated RNN outputs.

2.2 Feature maps extraction

Once the DCNN $f(\cdot,\ \boldsymbol{W}_\mathrm{DCNN})\colon \mathbb{R}^\mathrm{n} \to \mathbb{R}^{|\mathcal{C}|}$ has been fine-tuned, feature map extraction is performed. The network can be viewed as a composition of convolutional and fully connected layers, where pooling and nonlinear layers are omitted here for clarity. Feature map extraction is carried out by applying only the convolutional part of the network to each image $\boldsymbol{I}_\mathrm{k}$ belonging to the test query and reference sets, that is, to all images in $\boldsymbol{\mathcal{Q}}_\mathbf{test} \cup \boldsymbol{\mathcal{D}}_\mathbf{test}$. The resulting output $\boldsymbol{z}_\mathrm{k}$ corresponds to the convolutional feature maps associated with image $\boldsymbol{I}_\mathrm{k}$, and is referred to throughout this work as the image representation. Note that the RNN is used only during training to incorporate sequential context into the gradient updates that shape the convolutional filters. At inference, the RNN is discarded and only the fine-tuned convolutional layers are applied, keeping the extraction cost identical to that of a standard DCNN.

ACCEPTED

2.3 Place matching

In the last step we perform place matching. In general, one can ignore the data sequentiality and match a query image with the most similar reference image. Such matching is usually called the naïve algorithm, since it is easy to implement but "produces considerable false positives [...]" [14]. However, since the evaluation datasets were captured during vehicle traversals, we can also use sequence-based algorithms, which exploit the local neighborhood of the place captured by an on-board camera. We briefly describe three sequence-based matching algorithms used in our evaluation: SeqSLAM [5], cone-based SeqSLAM [30], and NOSeqSLAM [31].

Let $D \in \mathbb{R}^{|\boldsymbol{\mathcal{Q}}|\times|\boldsymbol{\mathcal{D}}|}$ denote the difference matrix, where each entry $D[j,i]$ is the sum of absolute element-wise differences between image representations $\boldsymbol{z}_{q_i}$ and $\boldsymbol{z}_{\boldsymbol{d}_j}$. A contrast-enhanced version $\widehat{D}$ is obtained by column-wise normalization as described in [5]. The correspondence score between the query image $\boldsymbol{I}_{q_i}$ and the reference image $\boldsymbol{I}_{d_j}$ is defined as

$$S_{j,i} = \min_{v} \sum_{t=i-\left\lfloor \frac{d_s}{2} \right\rfloor}^{i+\left\lfloor \frac{d_s}{2} \right\rfloor} \widehat{D}[j + v(t-i), t]. \tag{5}$$

The index sequence $(j + v(t-i), t)$ forms a linear trajectory in the difference matrix parameterized by velocity v, and a smaller $S_{j,i}$ indicates a better match. SeqSLAM is not limited to difference-based measures; any similarity metric such as cosine similarity over an association matrix A can be used by maximizing instead of minimizing. Cone-based SeqSLAM relaxes the linear trajectory assumption. For each query image in the local window, it identifies the best-matching reference image and checks whether that match falls within a cone defined by velocity bounds $v_{min} \leq v \leq v_{max}$. The correspondence score $S_{j,i}$ is the fraction of matches that satisfy this constraint.

In NOSeqSLAM, the space of admissible sequences is defined by d_s and a branching factor η. For a given pair $(\boldsymbol{I}_{q_i}, \boldsymbol{I}_{d_j})$, a directed acyclic graph (DAG) is constructed with this pair as the root. The root has η children $\{(\boldsymbol{I}_{q_{i-1}}, \boldsymbol{I}_{d_j}), \ldots, (\boldsymbol{I}_{q_i}, \boldsymbol{I}_{d_{j-\eta+1}})\}$; each child expands recursively until a depth of $\lfloor d_s/2 \rfloor$ is reached, forming the left subgraph. A symmetric right subgraph is built forward in time. Edge weights are defined as $w = 1 - A[n,m]$, where A is the association matrix, so that the shortest path corresponds to the sequence of highest similarity. Let $l^*_{i,j}$ and $r^*_{i,j}$ denote the vertex indices along the shortest paths from the root to leaves of the left and right subgraphs, respectively. The correspondence score is

$$S_{j,i} = A[j,i] + \sum_{(k,l)\in l^*_{i,j}} A[l,k] + \sum_{(k,l)\in r^*_{i,j}} A[l,k]. \tag{6}$$

NOSeqSLAM generalizes SeqSLAM: instead of restricting admissible sequences to linear trajectories, it allows arbitrary paths via shortest-path optimization, capturing nonlinear temporal correlations such as variations in traversal speed. Based on the achieved results, NOSeqSLAM is selected as the sequence-based matching method in our approach. For algorithmic details, we refer the reader to [34].

3 Experimental results

3.1 Datasets

The experiments were first conducted on the Oxford RobotCar dataset [32], frequently used in autonomous vehicles research. We constructed 4 sequences of images from 4 recorded traversals[1]. We picked the first 1000 GPS readings, each at least 0.5 m in radius with respect to a previous reading, from the first sequence. Other sequences were sampled in a similar manner, starting at most 3 m from the first position of the first sequence and ending at most 3 m from its last reading. Ground truths are constructed using a localization radius of 10 m. From the first to fourth, the sequences are assigned as $\boldsymbol{\mathcal{Q}}_{\text{train}}$, $\boldsymbol{\mathcal{D}}_{\text{train}}$, $\boldsymbol{\mathcal{Q}}_{\text{test}}$, and $\boldsymbol{\mathcal{D}}_{\text{test}}$, respectively.

[1] 2014-12-09 12:21:02 GMT, 2014-12-10 18:10:50 GMT, 2015-05-19 14:06:38 GMT (stereo/centre) and 2014-12-16 18:44:24 GMT (stereo/left).

ACCEPTED

Nordland is the second dataset used for both training and evaluation. While viewpoint variation is minimal due to the fixed train trajectory, substantial appearance changes arise from seasonal variations. We selected the first 1000 locations from the test/*/section1 directories of the Partitioned Nordland dataset [33]. The winter and summer subsets are used as $\boldsymbol{Q}_{\text{train}}$ and $\boldsymbol{\mathcal{D}}_{\text{train}}$, respectively, while the spring and fall subsets serve as $\boldsymbol{Q}_{\text{test}}$ and $\boldsymbol{\mathcal{D}}_{\text{test}}$.

Note that no separate validation split is employed. Training (softmax fine-tuning on $\boldsymbol{Q}_{\text{train}} \cup \boldsymbol{\mathcal{D}}_{\text{train}}$) and evaluation (place retrieval on $\boldsymbol{Q}_{\text{test}}$, $\boldsymbol{\mathcal{D}}_{\text{test}}$) are disjoint tasks performed on separate traversals. Hyperparameters are selected via grid search over the test-set retrieval performance, which introduces no label leakage since the softmax training objective and the retrieval evaluation metric operate on entirely different data and criteria.

3.2 Architecture selection, training, and feature extraction

We evaluated AlexNet [10], VGG16 [35], and ResNet50 [36] as DCNN backbones, together with Elman RNN [22], LSTM [23], and GRU [24]. Optimization was performed using stochastic gradient descent over 200 epochs, varying the learning rate $lr \in \{0.0005 + 0.00005 \cdot k\}_{k=0}^{15}$ and scheduler step $s \in \{10, 20\}$, with fixed momentum $m = 0.9$, weight decay $wd = 0.0$, and scheduler gamma $s_\gamma = 0.1$. Although non-zero weight decay is commonly used to promote generalization, we empirically observed that the best performance in our setting is achieved with $wd = 0.0$. Based on empirical results, ResNet50 was selected as the DCNN backbone, and a bidirectional LSTM was employed as the RNN architecture. The sequence length was varied as $L \in \{3,5\}$, and the number of hidden layers was set to $n_h \in \{1, 2\}$.

All images were preprocessed by resizing the shorter dimension to 256 px and applying a 224 × 224 px center crop. Feature maps from the fine-tuned ResNet50 were extracted up to and including its 49th convolutional layer. For comparison, feature maps from the conv3, conv4, conv5, and conv6 layers of HybridNet[15] were obtained using the publicly available Caffe implementation.

3.3 Quantitative evaluation

Evaluation is performed for each place matching method – naïve, SeqSLAM, cone-based SeqSLAM, and NOSeqSLAM – and each image representation – original ResNet50, fine-tuned ResNet50, and HybridNet. Sequence lengths of $d_s \in \{3,5,7,9,11,31,43,51\}$ are evaluated. A localization radius of ± 3 indices is used for match proposal [13], while score thresholding [30] is applied to handle poor matches. The corresponding precision-recall curves, ordered by AUC, are shown in Figure 3.

By analysing Figure 3 and Tables 1 and 2, we observe that on the Oxford dataset the fine-tuned representation coupled with NOSeqSLAM achieves the highest AUC (0.939), with all fine-tuned configurations occupying the top four positions and significantly outperforming both HybridNet (best 0.602) and the original ResNet50 (best 0.225). In Table 1, the four fine-tuned configurations use $L = 5$, $n_h = 1$, $s = 10$, with learning rates 0.00095, 0.00075, 0.00075, and 0.00065, respectively. In Table 2, all configurations use $L = 3$ and $s = 20$, except the third ($s = 10$), with $n_h = 1$ for the first three and $n_h = 2$ for the fourth, and learning rates 0.001, 0.001, 0.00125, and 0.0009, respectively. The large margin over HybridNet is likely because it was fine-tuned on scene-centric data rather than road traffic scenes. On the Nordland dataset, HybridNet and our fine-tuned representation both achieve near-perfect AUC above 0.99, which is attributable to the simpler matching task: the fixed train trajectory eliminates viewpoint variation, leaving only seasonal appearance change. Importantly, across both datasets and all matching algorithms, fine-tuning consistently improves over the original representation, confirming that the benefit generalizes across matching strategies.

4 Conclusion

In this paper, we proposed a visual place recognition approach based on a softmax fine-tuned DCNN extended with an RNN. Experimental results on the Oxford RobotCar and Nordland datasets show that the fine-tuned representation, coupled with NOSeqSLAM, consistently achieves the highest AUC. Moreover, fine-tuning improves performance regardless of the matching algorithm, confirming that the benefit generalizes across matching strategies. Future work will explore lightweight architectures and self-supervised fine-tuning to reduce the dependence on ground-truth correspondences.

ACCEPTED

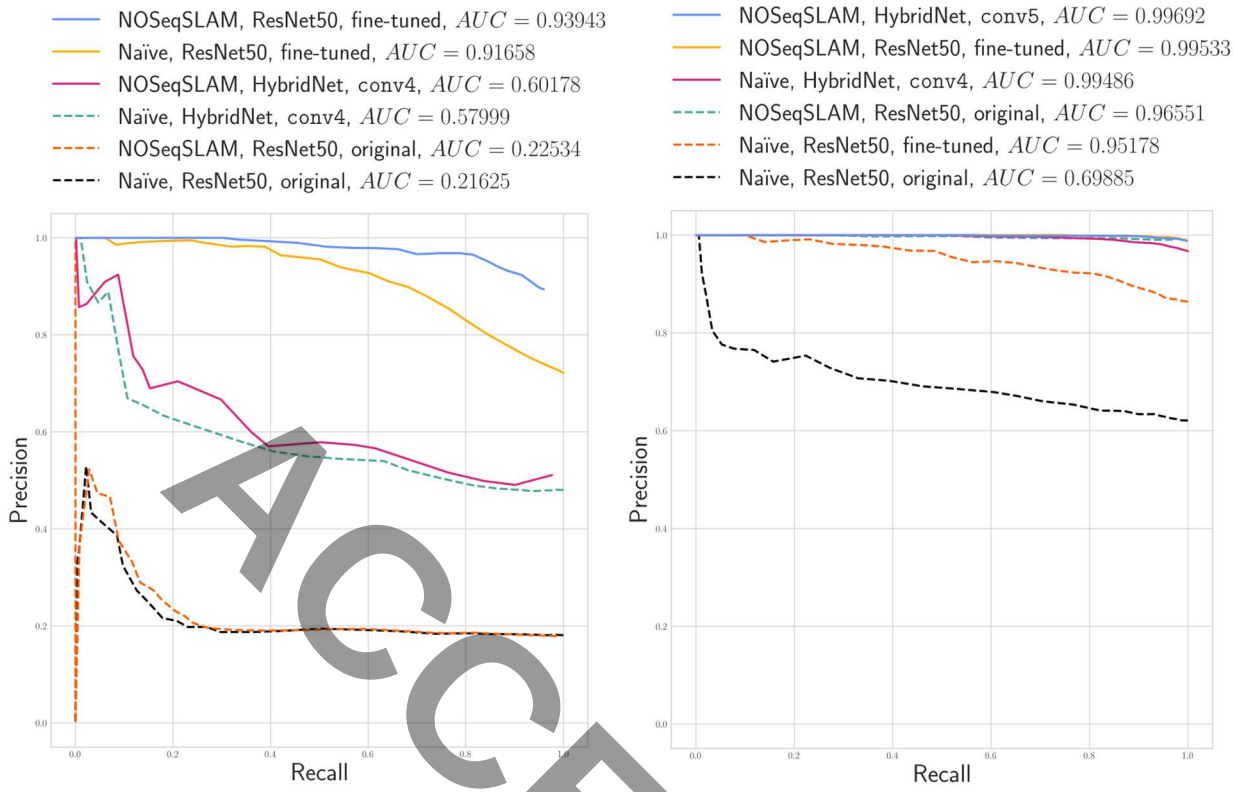


Figure 3. Precision-recall curves: (left) the Oxford and (right) the Nordland dataset

Table 1. AUC results, the Oxford RobotCar dataset.

Algorithm	Representation	AUC
NOSeqSLAM, $\boldsymbol{d_s = 31, \eta = 3}$	ResNet50, fine-tuned	0.93943
SeqSLAM, $\boldsymbol{d_s = 7}$	ResNet50, fine-tuned	0.92077
Naïve	ResNet50, fine-tuned	0.91658
SeqSLAM (cone), $\boldsymbol{d_s = 31}$	ResNet50, fine-tuned	0.82786
NOSeqSLAM, $\boldsymbol{d_s = 11, \eta = 3}$	HybridNet, conv4	0.60178
SeqSLAM, $\boldsymbol{d_s = 5}$	HybridNet, conv4	0.58190
Naïve	HybridNet, conv4	0.57999
SeqSLAM (cone), $\boldsymbol{d_s = 11}$	HybridNet, conv5	0.53417
NOSeqSLAM, $\boldsymbol{d_s = 3, \eta = 3}$	ResNet50, original	0.22534
SeqSLAM, $\boldsymbol{d_s = 3}$	ResNet50, original	0.22495
Naïve	ResNet50, original	0.21625
SeqSLAM (cone), $\boldsymbol{d_s = 11}$	ResNet50, original	0.20581

Table 2. AUC results, the Nordland dataset.

Algorithm	Representation	AUC
NOSeqSLAM, $\boldsymbol{d_s = 3, \eta = 3}$	HybridNet, conv5	0.99692
SeqSLAM, $\boldsymbol{d_s = 3}$	HybridNet, conv5	0.99612
NOSeqSLAM, $\boldsymbol{d_s = 5, \eta = 3}$	ResNet50, fine-tuned	0.99533
Naïve	HybridNet, conv4	0.99486
SeqSLAM, $\boldsymbol{d_s = 5}$	ResNet50, fine-tuned	0.99480
SeqSLAM (cone), $\boldsymbol{d_s = 51}$	ResNet50, fine-tuned	0.96809
NOSeqSLAM, $\boldsymbol{d_s = 31, \eta = 3}$	ResNet50, original	0.96551
SeqSLAM, $\boldsymbol{d_s = 31}$	ResNet50, original	0.95513
Naïve	ResNet50, fine-tuned	0.95178

ACCEPTED

SeqSLAM (cone), $\boldsymbol{d_s = 51}$	HybridNet, conv6	0.95000
SeqSLAM (cone), $\boldsymbol{d_s = 51}$	ResNet50, original	0.94471
Naïve	ResNet50, original	0.69885

5 Acknowledgements

Domagoj Matijević and Jurica Maltar acknowledge the support of the institutional research project "Advanced Algorithms and Optimization Models Supported by Mathematical Theory – OptimaAI", funded by the National Recovery and Resilience Plan 2021–2026 (NPOO) and implemented at the School of Applied Mathematics and Informatics, Josip Juraj Strossmayer University of Osijek.

References

[1] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual Place Recognition: A Survey," IEEE Trans. Robot., vol. 32, no. 1, pp. 1–19, 2016, doi: 10.1109/TRO.2015.2496823.

[2] O. Vysotska and C. Stachniss, "Lazy Data Association for Image Sequences Matching under Substantial Appearance Changes," IEEE Robot. Autom. Lett., vol. 1, no. 1, pp. 213–220, 2016, doi: 10.1109/LRA.2015.2512936.

[3] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the Performance of ConvNet Features for Place Recognition," in Proc. IEEE/RSJ IROS, Hamburg, Germany, Sep. 2015, pp. 4297–4304, doi: 10.1109/IROS.2015.7353986.

[4] R. Arandjelović, P. Gronát, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN Architecture for Weakly Supervised Place Recognition," in Proc. CVPR, Las Vegas, NV, USA, Jun. 2016, pp. 5297–5307, doi: 10.1109/CVPR.2016.572.

[5] M. J. Milford and G. F. Wyeth, "SeqSLAM: Visual Route-Based Navigation for Sunny Summer Days and Stormy Winter Nights," in Proc. IEEE ICRA, St. Paul, MN, USA, May 2012, pp. 1643–1649, doi: 10.1109/ICRA.2012.6224623.

[6] S. Lowry and M. Milford, "Supervised and Unsupervised Linear Learning Techniques for Visual Place Recognition in Changing Environments," IEEE Trans. Robot., vol. 32, no. 3, pp. 600–613, 2016, doi: 10.1109/TRO.2016.2545711.

[7] D. Lowe, "Object Recognition from Local Scale-Invariant Features," in Proc. IEEE ICCV, Kerkyra, Greece, Sep. 1999, pp. 1150–1157, doi: 10.1109/ICCV.1999.790410.

[8] H. Jégou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid, "Aggregating Local Image Descriptors into Compact Codes," IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 9, pp. 1704–1716, 2012, doi: 10.1109/TPAMI.2011.235.

[9] R. Cupec, E. K. Nyarko, D. Filko, A. Kitanov, and I. Petrović, "Place Recognition Based on Matching of Planar Surfaces and Line Segments," Int. J. Robot. Res., vol. 34, no. 4–5, pp. 674–704, 2015, doi: 10.1177/0278364914548708.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in Proc. NIPS, Lake Tahoe, NV, USA, Dec. 2012.

[11] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning Deep Features for Scene Recognition Using Places Database," Adv. Neural Inf. Process. Syst., vol. 27, pp. 487–495, 2014.

[12] C. Szegedy, W. Liu, Y. Jia et al., "Going Deeper with Convolutions," in Proc. CVPR, Boston, MA, USA, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.

[13] 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 Proc. IEEE ICRA, Seattle, WA, USA, May 2015, pp. 2774–2779, doi: 10.1109/ICRA.2015.7139576.

[14] T. Naseer, M. Ruhnke, C. Stachniss, L. Spinello, and W. Burgard, "Robust Visual SLAM across Seasons," in Proc. IEEE/RSJ IROS, Hamburg, Germany, Sep. 2015, pp. 2529–2535, doi: 10.1109/IROS.2015.7353721.

[15] Z. Chen, A. Jacobson, N. Sünderhauf et al., "Deep Learning Features at Scale for Visual Place Recognition," in Proc. IEEE ICRA, Singapore, May 2017, pp. 3223–3230, doi: 10.1109/ICRA.2017.7989366.

[16] G. Berton, C. Masone, and B. Caputo, "Rethinking Visual Geo-localization for Large-Scale

ACCEPTED

Applications," in Proc. CVPR, New Orleans, LA, USA, Jun. 2022, pp. 4878–4888, doi: 10.1109/CVPR52688.2022.00483.
[17] G. Berton, G. Trivigno, B. Caputo, and C. Masone, "EigenPlaces: Training Viewpoint Robust Models for Visual Place Recognition," in Proc. ICCV, Paris, France, Oct. 2023, pp. 11080–11090, doi: 10.1109/ICCV51070.2023.01017.
[18] A. Ali-bey, B. Chaib-draa, and P. Giguère, "MixVPR: Feature Mixing for Visual Place Recognition," in Proc. WACV, Waikoloa, HI, USA, Jan. 2023, pp. 2998–3007, doi: 10.1109/WACV56688.2023.00301.
[19] N. Keetha, A. Mishra, J. Karhade, K. M. Jatavallabhula, S. Scherer, M. Krishna, and S. Garg, "AnyLoc: Towards Universal Visual Place Recognition," IEEE Robot. Autom. Lett., vol. 9, no. 2, pp. 1286–1293, 2024, doi: 10.1109/LRA.2023.3343602.
[20] S. Izquierdo and J. Civera, "Optimal Transport Aggregation for Visual Place Recognition," in Proc. CVPR, Seattle, WA, USA, Jun. 2024, pp. 17658–17668, doi: 10.1109/CVPR52733.2024.01672.
[21] S. Garg and M. Milford, "SeqNet: Learning Descriptors for Sequence-Based Hierarchical Place Recognition," IEEE Robot. Autom. Lett., vol. 6, no. 3, pp. 4305–4312, 2021, doi: 10.1109/LRA.2021.3067633.
[22] J. L. Elman, "Finding Structure in Time," Cogn. Sci., vol. 14, no. 2, pp. 179–211, 1990, doi: 10.1207/s15516709cog1402_1.
[23] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.
[24] K. Cho, B. van Merriënboer, C. Gulcehre et al., "Learning Phrase Representations Using RNN Encoder–Decoder for Statistical Machine Translation," in Proc. EMNLP, Doha, Qatar, Oct. 2014, doi: 10.3115/v1/D14-1179.
[25] Y. Yu, X. Si, C. Hu, and J. Zhang, "A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures," Neural Comput., vol. 31, no. 7, pp. 1235–1270, 2019, doi: 10.1162/neco_a_01199.
[26] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network," Physica D Nonlinear Phenom., vol. 404, p. 132306, 2020, doi: 10.1016/j.physd.2019.132306.
[27] M. Chancán and M. Milford, "Sequential Place Learning: Heuristic-Free High-Performance Long-Term Place Recognition," arXiv:2103.02074, 2021, doi: 10.48550/arXiv.2103.02074.
[28] F. Lu, L. Zhang, X. Lan, S. Dong, Y. Wang, and C. Yuan, "Towards Seamless Adaptation of Pre-trained Models for Visual Place Recognition," in Proc. ICLR, Vienna, Austria, May 2024, doi: 10.48550/arXiv.2402.14505.
[29] M. Cummins and P. Newman, "FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance," Int. J. Robot. Res., vol. 27, no. 6, pp. 647–665, 2008, doi: 10.1177/0278364908090961.
[30] B. Talbot, S. Garg, and M. Milford, "OpenSeqSLAM2.0: An Open Source Toolbox for Visual Place Recognition under Changing Conditions," in Proc. IEEE/RSJ IROS, Madrid, Spain, Oct. 2018, pp. 7758–7765, doi: 10.1109/IROS.2018.8593761.
[31] J. Maltar, I. Marković, and I. Petrović, "Visual Place Recognition Using Directed Acyclic Graph Association Measures and Mutual Information-Based Feature Selection," Robot. Auton. Syst., vol. 132, p. 103598, 2020, doi: 10.1016/j.robot.2020.103598.
[32] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 Year, 1000km: The Oxford RobotCar Dataset," Int. J. Robot. Res., vol. 36, no. 1, pp. 3–15, 2017, doi: 10.1177/0278364916679498.
[33] D. Olid, J. M. Fácil, and J. Civera, "Single-View Place Recognition under Seasonal Changes," in Proc. PPNIV Workshop at IEEE/RSJ IROS, Madrid, Spain, Oct. 2018, doi: 10.48550/arXiv.1808.06516.
[34] J. Maltar, "Robust Visual Place Recognition Using Deep Representations and Sequence-Based Image Matching," Ph.D. dissertation, Univ. of Zagreb, Zagreb, Croatia, 2024.
[35] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv:1409.1556, 2015, doi: 10.48550/arXiv.1409.1556.
[36] K. He, X. Zhang, X. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. CVPR, Las Vegas, NV, USA, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

ACCEPTED