Matched Filtering Based LiDAR Place Recognition for Urban and Natural Environments

Therese Joseph, Tobias Fischer, *Senior Member, IEEE*, and Michael Milford, *Senior Member, IEEE*

***Abstract*—Place recognition is an important task within autonomous navigation, involving the re-identification of previously visited locations from an initial traverse. Unlike visual place recognition (VPR), LiDAR place recognition (LPR) is tolerant to changes in lighting, seasons, and textures, leading to high performance on benchmark datasets from structured urban environments. However, there is a growing need for methods that can operate in diverse environments with high performance and minimal training. In this letter, we propose a handcrafted matching strategy that performs roto-translation invariant place recognition and relative pose estimation for both urban and unstructured natural environments. Our approach constructs Birds Eye View (BEV) global descriptors and employs a two-stage search using matched filtering — a signal processing technique for detecting known signals amidst noise. Extensive testing on the NCLT, Oxford Radar, and WildPlaces datasets consistently demonstrates state-of-the-art performance across place recognition and relative pose estimation metrics, with up to 15% higher recall than previous state-of-the-art.**

***Index Terms*—Localization, range sensing.**

I. Introduction

IN AUTONOMOUS navigation, place recognition is the task of global relocalisation given a reference set of previously visited places. This capability is crucial for several downstream tasks in Simultaneous Localisation and Mapping (SLAM), such as loop closure detection and localisation in GPS-denied environments [1], [2]. Much of the research has focused on visual place recognition (VPR), which relies on features extracted from vision-based sensors to distinguish different locations, as demonstrated in methods like FAB-MAP [3], SeqSLAM [4], and NetVLAD [5]. Since visual features are sensitive to changes in lighting, weather and viewpoints; VPR methods rely on diverse training data that captures varied conditions to handle these changes. In contrast, geometric features captured by LiDAR sensors are inherently tolerant to variations in lighting, weather and seasons. Consequently, the field has seen the emergence of LiDAR Place Recognition (LPR), which utilises handcrafted or learned methods to transform long-range point cloud data with 360° field of view into efficient representations of a place [6], [7], [8].

While trained LPR methods can leverage large datasets to generate robust, efficient solutions, they often suffer from performance degradation in novel, unseen environments [9]. On the other hand, handcrafted methods, which do not require extensive training data, may struggle in diverse environments depending on the design of descriptors and tuning of parameters. The ideal LPR system should consistently perform well across varied environments with minimal training and fine-tuning. Motivated by this challenge, we propose a robust handcrafted matching strategy from signal processing called matched filtering – where detecting a known signal in the presence of noise closely parallels the place recognition problem. By leveraging a handcrafted Bird's Eye View (BEV) descriptor, combined with a global and local search architecture, our method effectively matches previously visited places with the current query, even in noisy and variable conditions as illustrated in Fig. 1.

Another challenge for LPR methods is achieving both rotational and translational invariance when positional discrepancies exist between the reference and query traverse. While some methods address this by generating roto-translation (rotation and translation) invariant descriptors [10], [11], [12], this invariance often comes at the cost of losing the capability to directly estimate relative pose – a critical downstream task in applications like loop closure in SLAM. To bridge this gap, some LPR methods apply point cloud registration techniques like ICP [13] and RANSAC [14] after place recognition to refine the scan alignment and estimate relative pose [10], [15], [16]. In contrast, our approach streamlines this process by directly estimating the relative pose from the matched filter output.

LPR methods are mostly validated on widely recognised urban driving datasets like Oxford RobotCar [17], KITTI [18], MulRan [19], and NCLT [20], where they demonstrate high performance. While these datasets are useful for benchmarking, they lack environmental diversity, as they are predominantly captured in structured, urban on-road settings. This focus on urban environments presents another limitation: LPR methods may overfit urban characteristics, such as regular geometric features, sparse occupancy and grid-like street layouts. Consequently, their performance will likely degrade in natural environments with irregular geometric features, seasonally varied vegetation and densely occupied space, as observed in [9]. Therefore, evaluating LPR methods in unstructured natural environments

Received 3 September 2024; accepted 7 January 2025. Date of publication 27 January 2025; date of current version 5 February 2025. This article was recommended for publication by Associate Editor S. Scherer and Editor S. Behnke upon evaluation of the reviewers' comments. The work of Michael Milford was supported in part by ARC Laureate Fellowship under Grant FL210100156, and in part by Intel Labs. The work of Tobias Fischer was supported in part by Intel Labs, and in part by ARC DECRA Fellowship under Grant DE240100149. This work was supported by the Queensland University of Technology (QUT) through the Centre for Robotics. *(Corresponding author: Therese Joseph.)*

The authors are with the QUT Centre for Robotics, School of Electrical Engineering and Robotics, Queensland University of Technology, Brisbane QLD 4000, Australia (e-mail: t2.joseph@hdr.qut.edu.au).

Digital Object Identifier 10.1109/LRA.2025.3533966

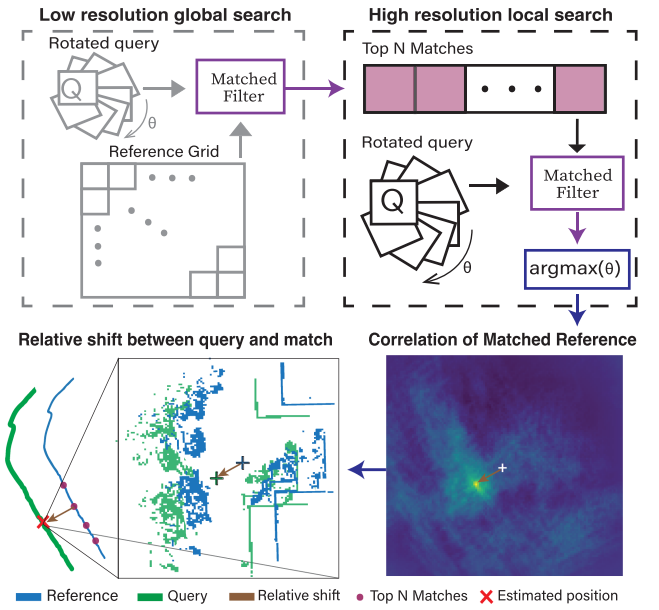

Fig. 1. Matched filter-based LiDAR Place Recognition (LPR) architecture. Low-resolution reference BEV descriptors are accumulated within a grid while the query BEV descriptor is rotated for a global search using the matched filter. The top n matches from this search are then used for a high-resolution local search. Finally, the resulting correlation output from the top match with maximum rotation alignment is used for pose correction from reference to query traverse.

like WildPlaces [21] is crucial for assessing their generalisation capabilities and promoting the development of robust, environment-invariant place descriptors.

In this letter, we aim to address these challenges in the field with the following contributions:

1) A two-stage architecture utilising low and high-resolution Birds Eye View (BEV) descriptors for efficient global and local search.
2) A matching strategy derived from signal processing, using matched filtering to achieve rotation and translation invariant LiDAR place recognition.
3) Direct relative pose estimation based on matched filtering outputs enabling precise relocalisation.
4) Extensive experiments on urban and natural environments consistently demonstrate SoTA performance in place recognition and pose estimation, with up to 15% higher recall than previous SoTA.

We publicly release our code at https://github.com/theresejoseph/Matched_Filter_based_LPR.

II. Related Works

In this section, we review the current literature to highlight the novelty and contributions of our method, situating it within the broader LPR research landscape. Sections II-A and II-B discuss LPR methods based on handcrafted and learned techniques, respectively. We further explore LPR methods that have been evaluated in unstructured environments in Section II-C. Lastly, in Section II-D, we examine the use of correlation-based matching strategies which are similar to our approach using matched filtering.

A. Handcrafted Methods for LPR

A primary goal of place recognition is to generate descriptors that uniquely and efficiently represent a location. These descriptors can either describe the entire scene globally or extract local distinctive features. Handcrafted methods achieve this by encoding geometric information such as density, height, orientation, depth, or intensity from LiDAR scans. For instance, 3D local descriptors like ISHOT [22] store histograms of surface normals, orientation and intensity; while 3D global descriptors like M2DP [11] store signatures of intensity from 2D sliced projections.

Some methods also transformed 3D point clouds into 2D projections with BEV or spherical range images to reduce dimensionality. For example, scan context-based methods [7], [23] use polar conversion of BEV as a global place descriptor; while LiDAR Iris [24] creates a binary encoding of BEV polar images with Log-Gabor filters to extract distinctive features. These BEV-based descriptors use circular shifting or rotational search to achieve rotational invariance, similar to our method. Conversely, BVmatch [15] applies Log-Gabor filters with a maximum index map to encode orientation information and achieve direct orientation invariance.

B. Learning-Based Methods for LPR

The rise of deep learning has also influenced more data-driven approaches to LPR. PointNetVLAD [25] was a pioneering method, utilising triplet and quadruplet loss functions to train on raw 3D point cloud data end-to-end. This approach was further refined by PCAN [26], which introduced an attention layer to better aggregate task-relevant features, and by LPDnet [12], which employed a graph network for adaptive local feature extraction and neighbourhood aggregation. Similarly, DH3D [16] uses a Siamese network with point convolution to detect and describe local features, which are then aggregated into global descriptors.

Other deep learning methods have utilised discretised point clouds processed through 3D CNNs. Notable examples include MinkLoc3D [6], which incorporates a feature pyramid network, TransLoc3D [27], which employs a transformer network with multiple scale reception and attention mechanisms, and LoGG3D Net [28], which integrates a sparse U-Net architecture with consistency loss to improve performance.

Despite the high performance of these methods on standard urban datasets like Oxford RobotCar [17], KITTI [18], MulRan [19], and NCLT [20], their generalisation to unstructured environments remains a significant challenge as observed in [9].

C. LPR in Natural Environments

Place recognition within complex unstructured natural environments is crucial for deploying autonomous systems in real-world scenarios, and some recent LPR methods have focused on this challenge. For instance, [29] introduces a handcrafted global descriptor for unstructured orchards, showing generalisation on the Kitti dataset [18]. Since the orchard dataset and reproducible

code have not yet been publicly released, we omitted a comparison to this work in our letter.

On the other hand, the recent release of the WildPlaces dataset [21] facilitated LPR evaluation in two natural unstructured environments: Venman and Karawatha. The handcrafted descriptor in BTC [30] is tested on WildPlaces and has high performance in Venman, but the performance significantly degrades in Karawatha. Moreover, the test time adaptation approach in GeoAdapt [9] has better performance than standard approaches when trained on urban datasets and tested on WildPlaces, though it still underperforms compared to direct training on WildPlaces. In contrast, our work demonstrates consistent SoTA place recognition performance on the WildPlaces dataset with minimal parameter tuning for urban to natural environments.

D. Correlation Based Matching

Cross-correlation is the underlying technique for matched filters in signal processing. Correlation-based methods have also been applied in autonomous navigation, particularly in scan matching for aligning 2D laser scans, as demonstrated in [31] and [32]. In LPR, the Radon Sinogram (RING) method [33] uses circular cross-correlation for place recognition and orientation estimation, with RING++ [10] extending this to include 2D cross-correlation for translation estimation. Learning-based LPR methods have also incorporated correlation into their framework. For example, DiSCO [34] and OverlapNet [8] use correlation for orientation estimation, while DeepRING [35] employs it as a distance vector.

In comparison, our method directly applies cross-correlation with the BEV-transformed query and reference set using matched filtering, enabling both place recognition and relative pose estimation. These handcrafted descriptors are also generated with minimal fine-tuning across diverse environments.

III. Methodology

This section outlines our LPR methodology, beginning with an explanation of our two-stage matching architecture in Section III-A. Next, we discuss the BEV descriptor generation in Section III-B and the matched filtering search process in Section III-C, which are the key components that enable the effectiveness of our method. Finally, we describe the pose estimation calculations using relative shift based on the matched filter output in Section III-D.

A. Two Stage Matching Architecture

Our method employs a two-stage hierarchical search to estimate query pose: a low resolution global search followed by high resolution local search, as illustrated in Fig. 1. In this method, we use the same descriptors and matched filter output for both place recognition and pose correction, reducing pose alignment overhead and streamlining the localisation process.

Initially, we construct a lookup grid of low-resolution reference BEV descriptors. In the first global search phase, a low-resolution query BEV descriptor is matched against the reference grid across k rotation increments (Fig. 1 top left). The top n matches from this phase are obtained based on the maximum correlation.

In the subsequent local search phase, high-resolution query and top n reference descriptors are matched again, across k rotations (Fig. 1 top right). The best match from this second stage is then used to evaluate the relative shift and estimate the query's pose (Fig. 1 bottom). This two-stage architecture, combined with matched filtering enables roto-translation invariant place recognition and direct pose estimation.

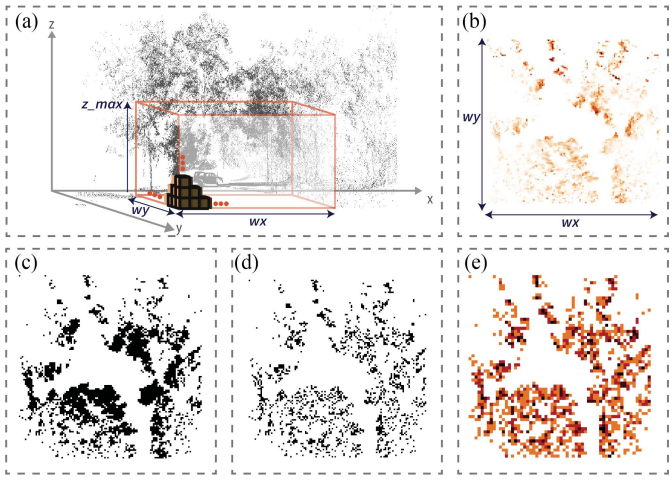

Fig. 2. Illustrated BEV descriptor generation process with a scan from the WildPlaces Dataset [21]: (a) Point cloud data cropped and voxelised. (b) BEV image created from a height density map. (c) Thresholded height density map forming a BEV descriptor indicating occupancy. (d) BEV descriptor after randomly downsampling patches. (e) Lower resolution BEV descriptor after average pooling.

B. BEV Descriptors

A BEV descriptor is generated from the LiDAR scans to represent each place uniquely. A point cloud from a single LiDAR scan is defined by $P = \{(x_i, y_i, z_i) \mid i = 1, 2, \ldots, N_p\}$, where (x_i, y_i, z_i) represents the coordinates of a point in 3D space. To reduce BEV occlusion from dense vertical structures like tree canopies, we crop the point cloud along the z-axis, retaining points up to $z_{\max}$. Next, a voxel grid is applied to partition the 3D space into uniform cubic volumes, each with a volume v. This grid is generated with a fixed number of x-y bins ($w_x \times w_y$) called cells, as illustrated in Fig. 2 panel a. This discretisation reduces noise and complexity, enabling a consistent scene representation.

The 3D point cloud is then projected into 2D by generating a BEV image with dimensions ($w_x \times w_y$), optimising search efficiency and reducing complexity (Fig. 2, panel b). This is achieved by constructing a height density map (HDM), representing the number of points along the z-axis within each cell (x-y bin) of the voxel grid.

This HDM BEV image is a standard approach used in LPR methods [10], [15], [34] and we extend our descriptor generation by including occupancy weight, patch downsampling and average pooling. In our descriptor, a threshold is applied to the

density map such that cells with a density greater than d are considered occupied, while all other cells are unoccupied with a predefined weighting w (Fig. 2 panel c), resulting in:

$$\mathbf{B}(i,j) = \mathbf{1}_{\{\mathbf{HDM}(i,j)>d\}} \cdot w \tag{1}$$

The BEV descriptors for the reference scans are also further down-sampled so that each $m \times m$ patch within the scan has at most c occupied cells. To achieve this, given the locations of all the occupied cells P in a patch $\mathbf{B}_{\text{patch}}$, a subset P' is randomly selected without replacement to include a maximum of c cells. The selected cells are considered occupied, while all others are marked as unoccupied with a predefined weighting w. The resulting patches $\mathbf{B}'_{\text{patch}}$ are combined to form the final downsampled descriptor, with p, q denoting the index of the cells within each patches:

$$P = \{(p,q) \mid \mathbf{B}_{\text{patch}}(p,q) = 1, 0 \le p < m, \quad 0 \le q < m\} \tag{2}$$

$$P' \subseteq P \quad \text{such that} \quad |P'| = \min(c, |P|) \tag{3}$$

$$\mathbf{B}'_{\text{patch}}(p,q) = \begin{cases} w & \text{if } (p,q) \in P' \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

This downsampling process is particularly useful for dense scans in the reference set, which might otherwise correlate with sparse scans and generate false matches. Fig. 2 panel d, illustrates how patch down-sampling maintains the global geometry while reducing the descriptor density to increase the distinction of environmental features.

Finally, average pooling is applied to the descriptors such that only the average of each $u \times u$ patch is stored, where p and q are indices for each patch. This generates the low-resolution BEV descriptors $\mathbf{B}_{\text{lowRes}}$ (Fig. 2 panel e) used for the global search:

$$\mathbf{B}_{\text{lowRes}}(i,j) = \frac{1}{u^2} \sum_{p=0}^{u-1} \sum_{q=0}^{u-1} \mathbf{B}(i \cdot u + p, j \cdot u + q). \tag{5}$$

C. Matched Filtering

Each search, from the two-stage architecture in Fig. 1, is conducted with a matched filter. In this process, the centre of each rotated input query $\mathbf{Q_{rot}}$ is shifted across each point in the reference grid $\mathbf{R}$ and correlated. The correlation function calculates the sum of products to measure the query similarity to the reference at every translation offset and rotation increment, defined by:

$$(\mathbf{Q} * \mathbf{R})(i_r, j_r, \theta) = \sum_{i_q} \sum_{j_q} \mathbf{Q}(i_q, j_q, \theta) \cdot \mathbf{R}(i_q + i_r, j_q + j_r, 0). \tag{6}$$

Correlation is equivalent to convolution with a vertically and horizontally reversed query representing the kernel. As a result, the matched filter is represented as a pointwise multiplication in the frequency domain according to the convolution theorem. Hence, the Fast Fourier Transform (FFT) algorithm is used to convert the matched filter inputs into the frequency domain and efficiently compute the filter outputs.

D. Pose Estimation

The matched filter output represents a distribution of match similarity across translation and rotation increments i_r, j_r, θ. Therefore, the maximum of this output corresponds to the position of the best match within the reference grid and the relative yaw variation between reference and query, as described by:

$$(i_{\text{match}}, j_{\text{match}}, \theta_{\text{match}}) = \arg\max_{i_r, j_r, \theta} ((\mathbf{Q} * \mathbf{R})(i_r, j_r, \theta)). \tag{7}$$

If the maximum correlation position is the centre of a scan in the reference grid, this suggests that the query scan was captured at the same location as the matched reference. However, in realistic datasets, position shifts often occur between the query and reference scans. In such cases, the matched filter output reflects this shift by having the match position offset from the centre by $i_{\text{match}} - i_{\text{center}}$ and $j_{\text{match}} - j_{\text{center}}$, as indicated in the bottom right of Fig. 1. By combining this relative shift information with the matched reference pose x_r, y_r, θ_r, the precise location of the query scan can be estimated with:

$$\begin{pmatrix} x_q \\ y_q \end{pmatrix} = \begin{pmatrix} x_r \\ y_r \end{pmatrix} + \begin{pmatrix} \cos(\theta_r) & -\sin(\theta_r) \\ \sin(\theta_r) & \cos(\theta_r) \end{pmatrix} \begin{pmatrix} i_{\text{match}} - i_{\text{center}} \\ j_{\text{match}} - j_{\text{center}} \end{pmatrix}. \tag{8}$$

IV. Experimental Setup

A. Datasets

We conducted experiments on three long-term, large-scale datasets from diverse structured and unstructured environments: NCLT [20], WildPlaces [21] and Oxford Radar RobotCar [17], demonstrated in Fig. 3.

1) NCLT [20] is a long-term urban driving dataset collected on the University of Michigan campus comprising 27 traversals over 15 months of a 5.5-kilometre route. It includes indoor and outdoor environments, multiple seasons, different times of day and various routes captured using a Velodyne HDL 32E LiDAR. The dataset also provides accurate ground truth collected using a NovAtel DL-4 with RTK GPS. Using 2 reference traverses from 2013 and 9 traverses from 2012, we tested our method on NCLT following the implementation in BVmatch [15], with query scans extracted every 10 m.

2) WildPlaces [21] is a large-scale dataset with 8 traverses collected in natural, unstructured outdoor environments from Venman and Karawatha in Brisbane, Australia. It covers 33 kilometers on various routes through two different environments (Venman and Karawatha) over 14 months. The scans were collected using a Velodyne VPL 16 LiDAR scanner with a range of 120 meters, mounted on a rotating brushless DC motor to achieve 120°vertical FoV. The submap scans of this dataset and the ground truth positioning information were obtained using WildCat SLAM which integrated the IMU measurements from a Microstrain 3DMCV5-25 9-DoF sensor and the raw LiDAR measurements. We compared our method to the baselines provided with this dataset using open-source code to determine the query evaluation split for their eight sequences.

3) The Oxford Radar RobotCar dataset [17] is a radar extension on the popular Oxford Robot Car collected in 2019.

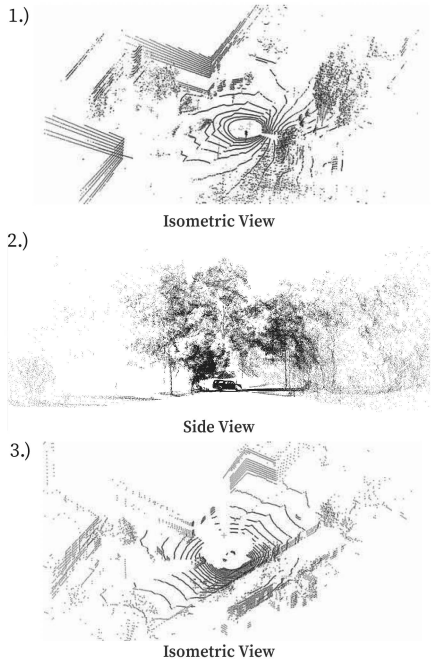

Fig. 3. Sample LiDAR scans from three different datasets: **1.)** NCLT Dataset scan using Velodyne 32E-HDL LiDAR sensor captured in an urban driving environment from Michigan [20]. **2.)** WildPlaces Dataset scan using Velodyne 16-VPL LiDAR sensor captured in unstructured natural environments from Brisbane [21]. **3.)** Oxford Radar Dataset scan using Velodyne32-VPL LiDAR sensor captured in an urban driving environment from Oxford [17].

It includes 32 traversals of a 9-kilometer route through urban central Oxford capturing various weather, traffic, and lighting conditions. The LiDAR scans were obtained using two Velodyne HDL-32E sensors, and we arbitrarily picked the left sensor for this investigation. It also comes with ground truth positioning information from NovAtel SPAN-CPT ALIGN INS/GPS sensor. Following BVmatch [15], we evaluate Oxford Radar query scans from every 10 m using our method with 2 reference traverses and 5 query traverses.

B. Evaluation Metrics

We employed a combination of place recognition and pose estimation metrics, following [10], [15], [21], to evaluate our method. We calculated recall @1 for all datasets by determining the percentage of true positives or estimated locations within a specified distance threshold. For WildPlaces, we used a threshold of 3 m, and for NCLT and Oxford Radar, we used a threshold of 25 m, following the comparison methods. This was calculated using:

$$\text{Recall @1} = \frac{TP}{TP + FP}. \tag{9}$$

Since our method predicts a 3DOF pose estimate, we evaluated the rotation and translation errors of the positive place recognition pairs based on recall @1. The relative translation error (RTE) was calculated as the Euclidean distance between the ground truth 2D position ρ_{GT} and the estimated position ρ_{est}: $\text{RTE} = ||\rho_{\text{GT}} - \rho_{\text{est}}||$.

The relative rotation error (RRE) is the absolute difference between the ground truth yaw γ_{GT} and the estimated yaw γ_{est}: $\text{RRE} = |\gamma_{\text{GT}} - \gamma_{\text{est}}|$.

An additional success rate criterion was measured for pose estimation, following BVmatch [15]. The success rate is the percentage of pose estimates with less than 2 m RTE and 5° RRE. This was formulated as:

$$\text{SR} = \frac{TP_{RTE<2m \,\wedge RRE<5^\circ}}{TP}. \tag{10}$$

C. Implementation Details

We selected and fine-tuned hyperparameters for our method with ablation in Section V-E. Key parameters including 120×120 cell for descriptor size (w_x, w_y), top 2 matches (n), 2×2 cells for the average pooling window (u), -0.15 unoccupied cell weighting (w), 10° rotation increment (k), 10×10 cells for the patch downsampling window (m) and 20 maximum occupied cells in a patch (c), were standardised across all experiments. Next, we chose voxel volumes (v) inversely proportional to the average descriptor density (ρ_{BEV}) such that $v = 0.001\rho_{\text{BEV}} + 1.7$. Here, the BEV descriptor density (ρ_{BEV}) was determined by creating descriptors with a fixed volume of 1 m^3 and counting the number of occupied cells. This resulted in voxel volumes of 0.3 m^3, 0.75 m^3 and 1.3 m^3 for WildPlaces, NCLT and Oxford Radar, respectively.

We also fine-tuned hyperparameters to adapt to the varying environmental conditions. The height of scans (z_{max}) in the WildPlaces dataset was cropped at 3 m to remove the tree foliage and extract the shape of the cleared path, a step unnecessary in urban environments. Additionally, a density threshold (d) 1 was used for urban and 2 for natural environments to account for higher density in forest environments.

V. Results and Discussion

This section presents the performance of our method across three datasets: Section V-A evaluates performance in natural environments following the implementation in WildPlaces [21], while Section V-B focuses on urban environments (NCLT and Oxford Radar RobotCar) following the implementation in BVmatch [15]. Section V-C compares our method with other correlation-based approaches. We assess our pose estimation capabilities in Section V-D and present an ablation study on hyperparameters in Section V-E. Finally, Section V-F provides a runtime evaluation of our descriptor generation and search architecture.

A. Place Recognition in Natural Environment

In the WildPlaces dataset, we conducted 24 experiments, comparing every combination of reference and query from 4 traverses in both Karawatha and Venman. The average recall of our method compared to [6], [7], [27], [28] is presented in Table I. We observe large performance improvements, with ∼15% higher recall @1 for both Venman and Karawatha sequences on average.

TABLE I
AVERAGE RECALL @1 ON WILDPLACES DATATSET

Method	Venman	Karawatha	Average
Scan Context [7]	33.98	38.44	36.21
TransLoc3D [27]	50.24	46.08	48.16
MinkLoc3Dv2 [6]	75.77	67.82	71.80
LoGG3D-Net [28]	79.84	74.67	77.26
Ours	**94.46**	**90.50**	**92.48**

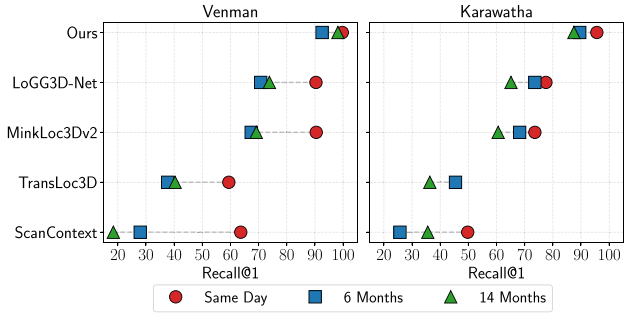

Fig. 4. Long Term Place Recognition Performance. Examination of LPR performance for query collected from day one, tested on references from the same day, 6 months later and 14 months later using 5 different methods, including ours.

Since the WildPlaces dataset was collected over 14 months, we also evaluated long-term place recognition to analyse the challenges arising from long-term changes in a dense forest. Using sequence 2 collected on June 2021 as a query, we measured recall @1 on sequence 1 (same day, June 2021), sequence 3 (6 months later, December 2021) and sequence 4 (14 months later, August 2022). These long-term sequences faced different challenges: winter-to-summer seasonal change over 6 months and longer-term variations over 14 months. Following the trend of the comparison methods, our method also exhibited the highest performance on the same day as illustrated in Fig. 4. We observed a 7% drop in recall after 6 months and an 8% drop after 14 months, compared to 20% and 12% drop in the best baseline for Venman and Karawatha, respectively. Our long-term place recognition at 14 months also outperforms the same-day performance of all comparison methods.

A possible reason for the improved performance is the additional BEV descriptor processing and voxel size fine-tuning which allowed our method to handle denser point clouds.

B. Place Recognition in Urban Environments

For the urban NCLT and Oxford radar datasets, we compare to a combination of handcrafted features and deep learning techniques [11], [12], [15], [16], [25], [26], following the results presented in BVmatch [15]. Using the performance metric recall @1, we outperform the baselines on both datasets with significant improvement, ∼10%, in NCLT and 0.7% in Oxford Radar, as presented in Table II. We note that although the baselines exhibit performance variations of around 10% across the two urban datasets, which were captured in similar environments

TABLE II
AVERAGE RECALL @1 ON NCLT & OXFORD RADAR

Method	Oxford Radar	NCLT
M2DP [11]	50.9	36.2
PN-VLAD [25]	86.6	62.8
PCAN [26]	80.3	59.0
LPD-Net [12]	90.0	72.5
DH3D [16]	78.2	59.4
BVMatch [15]	93.9	83.6
Ours	**94.6**	**92.9**

TABLE III
POSE ESTIMATION RESULTS ON NCLT

Method	RTE (m)		RRE (deg)		SR (%)
	Mean	Std.	Mean	Std.	
SIFT [36]	0.69	0.46	1.49	1.23	53.3
DH3D [16]	1.16	0.54	3.43	1.06	14.8
OverlapNet [8]	-	-	2.43	1.42	15.4
BVMatch [15]	0.57	0.38	**1.08**	**1.00**	94.5
Ours	**0.48**	**0.29**	1.43	1.07	**97.7**

(as illustrated in Fig. 3), our method demonstrates consistently high performance across both datasets.

C. Place Recognition With Correlation-Based Methods

We also compare the place recognition performance of our method with other correlation-based matching methods, specifically RING++ [10], which uses handcrafted features, and DeepRING [35], which employs learned feature extraction. While we initially aimed to evaluate these methods using the BVMatch evaluation split, we were unable to reproduce the reported results. Consequently, we reran our experiments using the evaluation splits from their respective papers and compared our performance to their reported results.

RING++ reports a recall @1 of 73.2% compared to 90.8% for our method. This was tested on the NCLT dataset using a revisit threshold of 10 meters, with the reference sequence "2012-02-04" sampled every 20 meters and the query sequence "2012-08-20" sampled every 5 meters. DeepRING, utilising the same reference sequence and sampling intervals but with the query sequence "2012-03-17," achieves a recall @1 of 85.9% whereas our method achieves 88.0%. Without any changes to our hyperparameters, our method outperforms both RING++ and DeepRING in terms of place recognition recall.

D. Pose Estimation

To evaluate our pose estimation method, we compared it against the results reported in BVmatch [15]. Performance was assessed using RTE, RRE, and SR metrics on a single reference-query pair from the NCLT dataset, as shown in Table III. Our method achieved 3.5% more successful matches and demonstrated 0.09 m lower mean and deviation in RTE for the correct poses. However, due to the coarse rotation search used in our approach, BVmatch, which incorporates ICP refinement, achieved a 0.35° lower mean and a 0.07° lower deviation for RRE. Ablation of the hyperparameters in Fig. 7 shows our

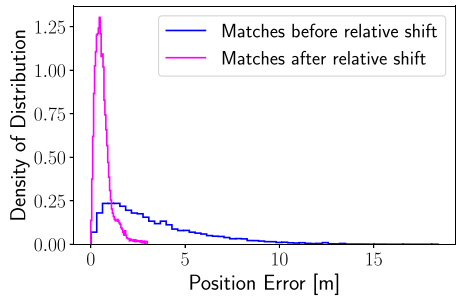

Fig. 5. Impact of the relative shift on position estimation for the complete WildPlaces dataset: The position error density distribution of correct LPR matches with and without relative pose correction demonstrates a lower mean error and smaller deviation after correction.

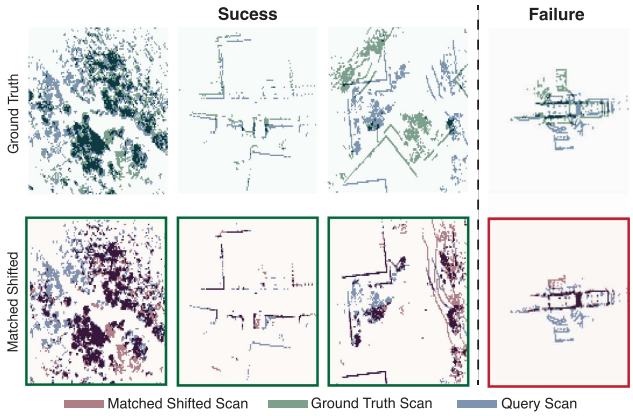

Fig. 6. A selection of matched scans with their corresponding ground truth overlaid onto the query. The scans were selected from WildPlaces (column 1), Oxford Radar (column 2), and NCLT (columns 3 and 4). A green outline around the first three scans indicates a successful match, while a red outline indicates an incorrect match.

performance can be improved with finer rotation increments, albeit at the cost of increased run time.

Unlike the comparison methods, the primary advantage of our pose estimation approach lies in its ability to directly estimate relative shifts from the matched filter output, which is then used to correct the reference pose. This advantage is evident in Fig. 5, which shows the distribution of position errors for successful matches in the WildPlaces dataset. After applying relative shift correction, the mean and standard deviation of pose errors were significantly reduced. Consequently, our roto-translation invariant place search not only enhances place recognition but also facilitates direct and accurate pose estimation.

The effectiveness of our pose estimation method can also be visualised using BEV descriptors. Fig. 6 illustrates a selection of query scans overlaid with ground truth and matched shifted scans, presented in two rows. Our method successfully identifies matches in dense outdoor environments and scenarios with significant translation and rotation variations in the ground truth scans. A failure case from the NCLT dataset is also shown, where an incorrect match occurred due to multiple aliased scans in the reference data along a narrow path. Additional failure scenarios were observed when significant feature changes occurred between the reference and query scans, when the coarse rotation

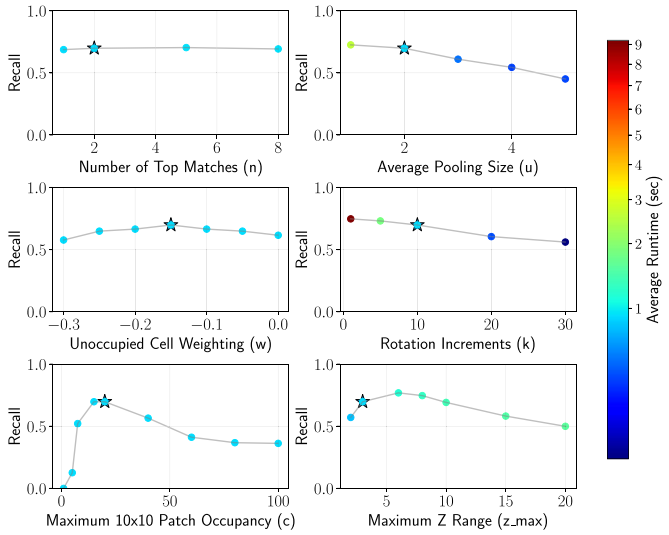

Fig. 7. Ablation of Parameters. Examination of recall on sequence 2 (reference) and 3 (query) from WildPlaces Venman Dataset while varying: number of top matches (n), weighting (w) for unoccupied cells in the BEV descriptor, average pooling size (u), rotation increment (k) for query scans, maximum number of occupied cells in a patch (c) and maximum point cloud height ($z_{\max}$). The black star indicates the chosen value for each hyperparameter while the colour of the points depicts runtime.

increment failed to align the query, and in environments with minimal distinguishable features.

E. Parameter Ablation

We also conducted an ablation study on the impact of key parameters on place recognition performance. As illustrated in Fig. 7, the hyperparameters chosen for our implementation maximised the tradeoff between performance and computational cost.

F. Runtime Evaluation

We implemented and evaluated our method on a desktop computer with an NVIDIA GeForce RTX 3090 GPU and an 11th Gen Intel Core i7 processor. The average descriptor generation times were approximately 30 ms for Oxford Radar, 40 ms for NCLT, and 300 ms for WildPlaces. The longer descriptor generation time for WildPlaces is attributed to the denser point clouds, which have more data points for processing. With 1500 reference scans, after subsampling every 2 meters, the average global search time is 460 ms. This scales linearly based on the number of reference scans. The local search with top 2 matches and pose estimation only adds 7 ms on average to the total runtime. Consequently, the overall method, including descriptor generation place recognition and pose estimation, operates at approximately 2 Hz (507 ms/query) for urban and 1 Hz (1 sec/query) for natural environments.

In comparison, the runtime of BVmatch in urban environments is 590 ms with the most computationally expensive step being descriptor generation and pose refinement with RANSAC. Another similar method for urban environments RING++ has an average runtime of 50 ms excluding pose refinement.

VI. Conclusion and Future Work

In this letter, we presented a two-stage, roto-translation invariant LiDAR place recognition method capable of direct relative pose estimation. Our method demonstrated SoTA performance in both urban and unstructured environments, with minimal fine-tuning. One limitation of our method is the assumption of minimal pitch and roll variations, which is unsuitable for platforms like drones and quadrupeds. Currently, our method is tailored for ground vehicles with 3DOF pose correction rather than 6DOF pose estimation. Additionally, the runtime could be optimised for longer traverses involving several thousand reference scans. To address these challenges, future work will focus on generating a comprehensive BEV map of the entire reference traverse, reducing search time and enhancing efficiency. This LPR method also holds potential for integration within a SLAM pipeline to enable loop closure detection in future works.

References

[1] S. Lowry et al., "Visual place recognition: A survey," *IEEE Trans. Robot.*, vol. 32, no. 1, pp. 1–19, Feb. 2016.

[2] S. Garg, T. Fischer, and M. Milford, "Where is your place, visual place recognition?," in *Proc. Int. Joint Conf. Artif. Intell.*, 2021, pp. 4416–4425.

[3] M. Cummins and P. Newman, "FAB-MAP: Probabilistic localization and mapping in the space of appearance," *Int. J. Robot. Res.*, vol. 27, pp. 647–665, 2008.

[4] M. J. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2012, pp. 1643–1649.

[5] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5297–5307.

[6] J. Komorowski, "Improving point cloud based place recognition with ranking-based loss and large batch training," in *Proc. Int. Conf. Pattern Recognit.*, 2022, pp. 3699–3705.

[7] G. Kim and A. Kim, "Scan context: Egocentric spatial descriptor for place recognition within 3D point cloud map," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 4802–4809.

[8] X. Chen et al., "OverlapNet: Loop closing for LiDAR-based SLAM," in *Proc. Robot., Sci. Syst.*, Jul. 2020, pp. 1–10.

[9] J. Knights, S. Hausler, S. Sridharan, C. Fookes, and P. Moghadam, "GeoAdapt: Self-supervised test-time adaptation in LiDAR place recognition using geometric priors," *IEEE Robot. Automat. Lett.*, vol. 9, no. 1, pp. 915–922, Jan. 2024.

[10] X. Xu et al., "RING++ : Roto-translation-invariant gram for global localization on a sparse scan map," *IEEE Trans. Robot.*, vol. 39, no. 6, pp. 4616–4635, Dec. 2023.

[11] L. He, X. Wang, and H. Zhang, "M2DP: A novel 3D point cloud descriptor and its application in loop closure detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 231–237.

[12] Z. Liu et al., "LPD-Net: 3D point cloud learning for large-scale place recognition and environment analysis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 2831–2840.

[13] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in *Proc. Sensor Fusion IV, Control Paradigms Data Structures*, 1992, pp. 586–606.

[14] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. Assoc. Comput. Machinery*, vol. 24, pp. 381–395, 1981.

[15] L. Luo, S.-Y. Cao, B. Han, H.-L. Shen, and J. Li, "BVMatch: Lidar-based place recognition using bird's-eye view images," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 6076–6083, Jul. 2021.

[16] J. Du, R. Wang, and D. Cremers, "DH3D: Deep hierarchical 3D descriptors for robust large-scale 6DoF relocalization," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 744–762.

[17] D. Barnes, M. Gadd, P. Murcutt, P. Newman, and I. Posner, "The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 6433–6438.

[18] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, pp. 1231–1237, 2013.

[19] G. Kim, Y. S. Park, Y. Cho, J. Jeong, and A. Kim, "Mulran: Multimodal range dataset for urban place recognition," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 6246–6253.

[20] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of Michigan North Campus long-term vision and lidar dataset," *Int. J. Robot. Res.*, vol. 35, pp. 1023–1035, 2016.

[21] J. Knights, K. Vidanapathirana, M. Ramezani, S. Sridharan, C. Fookes, and P. Moghadam, "Wild-places: A large-scale dataset for lidar place recognition in unstructured natural environments," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 11322–11328.

[22] J. Guo, P. V. K. Borges, C. Park, and A. Gawel, "Local descriptor for robust place recognition using LiDAR intensity," *IEEE Robot. Automat. Lett.*, vol. 4, no. 2, pp. 1470–1477, Apr. 2019.

[23] G. Kim, S. Choi, and A. Kim, "Scan context++ : Structural place recognition robust to rotation and lateral variations in urban environments," *IEEE Trans. Robot.*, vol. 38, no. 3, pp. 1856–1874, Jun. 2022.

[24] Y. Wang, Z. Sun, C.-Z. Xu, S. E. Sarma, J. Yang, and H. Kong, "LiDAR Iris for loop-closure detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5769–5775.

[25] M. A. Uy and G. H. Lee, "PointNetVLAD: Deep point cloud based retrieval for large-scale place recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4470–4479.

[26] W. Zhang and C. Xiao, "PCAN: 3D attention map learning using contextual information for point cloud based retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 12436–12445.

[27] T.-X. Xu, Y.-C. Guo, Z. Li, G. Yu, Y.-K. Lai, and S.-H. Zhang, "TransLoc3D: Point cloud based large-scale place recognition using adaptive receptive fields," *Commun. Inf. Syst.*, vol. 23, pp. 57–83, 2022.

[28] K. Vidanapathirana, M. Ramezani, P. Moghadam, S. Sridharan, and C. Fookes, "LoGG3D-Net: Locally guided global descriptor learning for 3D place recognition," in *Proc. Int. Conf. Robot. Automat.*, 2022, pp. 2215–2221.

[29] F. Ou, Y. Li, and Z. Miao, "Place recognition of large-scale unstructured orchards with attention score maps," *IEEE Robot. Autom. Lett.*, vol. 8, no. 2, pp. 958–965, Feb. 2023.

[30] C. Yuan, J. Lin, Z. Liu, H. Wei, X. Hong, and F. Zhang, "BTC: A binary and triangle combined descriptor for 3D place recognition," *IEEE Trans. Robot.*, vol. 40, pp. 1580–1599, 2024.

[31] E. B. Olson, "Real-time correlative scan matching," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2009, pp. 4387–4393.

[32] J. Konecny, M. Prauzek, P. Kromer, and P. Musilek, "Novel point-to-point scan matching algorithm based on cross-correlation," *Mobile Inf. Syst.*, vol. 2016, 2016, Art. no. 6463945.

[33] S. Lu, X. Xu, H. Yin, Z. Chen, R. Xiong, and Y. Wang, "One ring to rule them all: Radon sinogram for place recognition, orientation and translation estimation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 2778–2785.

[34] X. Xu, H. Yin, Z. Chen, Y. Li, Y. Wang, and R. Xiong, "DISCO: Differentiable scan context with orientation," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2791–2798, Apr. 2021.

[35] S. Lu, X. Xu, L. Tang, R. Xiong, and Y. Wang, "DeepRING: Learning roto-translation invariant representation for LiDAR based place recognition," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 1904–1911.

[36] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, pp. 91–110, 2004.