RadarSLAM: A robust simultaneous localization and mapping system for all weather conditions

Ziyang Hong, Yvan Petillot, Andrew Wallace and Sen Wang

Abstract
A Simultaneous Localization and Mapping (SLAM) system must be robust to support long-term mobile vehicle and robot applications. However, camera and LiDAR based SLAM systems can be fragile when facing challenging illumination or weather conditions which degrade the utility of imagery and point cloud data. Radar, whose operating electromagnetic spectrum is less affected by environmental changes, is promising although its distinct sensor model and noise characteristics bring open challenges when being exploited for SLAM. This paper studies the use of a Frequency Modulated Continuous Wave radar for SLAM in large-scale outdoor environments. We propose a full radar SLAM system, including a novel radar motion estimation algorithm that leverages radar geometry for reliable feature tracking. It also optimally compensates motion distortion and estimates pose by joint optimization. Its loop closure component is designed to be simple yet efficient for radar imagery by capturing and exploiting structural information of the surrounding environment. Extensive experiments on three public radar datasets, ranging from city streets and residential areas to countryside and highways, show competitive accuracy and reliability performance of the proposed radar SLAM system compared to the state-of-the-art LiDAR, vision and radar methods. The results show that our system is technically viable in achieving reliable SLAM in extreme weather conditions on the RADIATE Dataset, for example, heavy snow and dense fog, demonstrating the promising potential of using radar for all-weather localization and mapping.

Keywords
radar sensing, simultaneous localization and mapping, all-weather perception

1. Introduction
Simultaneous Localization and Mapping (SLAM) has attracted substantial interest over recent decades, and extraordinary progress has been made in the last 10 years in both the robotics and computer vision communities. In particular, camera and LiDAR based SLAM algorithms have been extensively investigated (Engel et al. (2014), Mur-Artal et al. (2015), Zhang and Singh (2014), Shan and Enlgot (2018)) and progressively applied to various real-world applications. Their robustness and accuracy are also improved further by fusing with other sensing modalities, especially Inertial Measurement Unit (IMU) based motion as a prior (Qin et al. (2018), Campos et al. (2020), Shan et al. (2020)).

Most existing camera and LiDAR sensors fundamentally operate within or near-visible electromagnetic spectra, which means that they are more susceptible to illumination changes, floating particles and water drops in the environments. It is well-known that vision suffers from low illumination, causing image degradation with dramatically increased motion blur, pixel noise and texture losses. The quality of LiDAR point clouds and camera images can also degenerate significantly, for instance, when facing a realistic density of fog particles, raindrops and snowflakes in mist, rain and snow. Studies of the degradation of LiDAR sensors are performed in (Jokela et al. (2019); Carballo et al. (2020)) and suggest that the data of all the tested LiDAR sensors degrades, to some extent, in foggy, rainy and snowy conditions. Given the fact that a motion prior is mainly effective in addressing short-period and temporary sensor degradation, even visual-inertial or LiDAR-inertial SLAM systems are anticipated to fail in these challenging weather conditions.

Radar is a type of active sensor, whose electromagnetic spectrum usually lies in a much lower frequency (GHz) band than camera and LiDAR (from THz to PHz). Therefore, it can operate more reliably in the majority of weather and light conditions. It also has additional benefits, for example,
further sensing range, relative velocity estimates from the Doppler effect and absolute range measurement. Recently, radar has been gradually considered to be indispensable for safe autonomy and has been increasingly adopted in the automotive industry for obstacle detection and Advanced Driver-Assistance Systems (ADAS). Meanwhile, recent advances in Frequency-Modulated Continuous-Wave (FMCW) radar systems make radar sensing more appealing since it is able to provide a relatively dense representation of the environment, instead of only returning sparse detections. With these advancements in radar sensing, radar-based SLAM system can be deployed on various platforms and in different environments, for example, surface mining, underground mining, off-road driving.

However, radar has a distinct sensor model and its data is formed very differently from vision and LiDAR. There are different challenges for radar-based SLAM compared to vision and LiDAR based SLAM. For example, its noise and clutter characteristics are complex, for example, electromagnetic radiation in the atmosphere and multi-path reflection, and its noise level tends to be much higher. This means that existing feature extraction and matching algorithms may not be well suited for radar images. Unlike LiDAR sensors, most FMCW radars do not provide 3D elevation information.

In this paper, we propose a novel SLAM system based on a FMCW radar. It can operate in various outdoor scenarios, for example, busy city streets and highways, and weather conditions, for example, heavy snowfall and dense fog, see Figure 1. Our main contributions are:

- A robust data association and outlier rejection mechanism for radar-based feature tracking by leveraging radar geometry.
- A novel motion compensation model formulated to reduce motion distortion induced by a low scanning rate. The motion compensation is jointly optimized with pose estimation in an optimization framework.
- A fast and effective loop closure detection scheme designed for a FMCW radar with dense returns.
- Extensive experiments on three available public radar datasets, demonstrating and validating the feasibility of a SLAM system operating in extreme weather conditions.
- Unique robustness and minimal parameter tuning, that is, the proposed radar SLAM system is the only competing method which can work properly on all data sequences, in particular using an identical set of parameters without much parameter tuning.

The rest of the paper is structured as follows. In Sec. 2, we discuss related work. In Sec. 3, we elaborate on the geometry of radar sensing and the challenges of using radar for SLAM. The proposed motion compensation tracking model is presented in Sec. 4, followed by the loop closure detection and pose graph optimization in Sec. 5. Experiments, results and system parameters are presented in Sec. 6. Finally, the conclusions and future work are discussed in Sec. 7.

2. Related work

In this section, we discuss related work on localization and mapping in extreme weather conditions using optical sensor modalities, that is, camera and LiDAR. We also review the past and current state-of-the-art radar-based localization and mapping methods.
2.1. Vision and LiDAR based localization and mapping in adverse weathers

Typical adverse weather conditions include rain, fog and snow which usually cause degradation in image quality or produce undesired effects, for example, due to rain streaks or ice. Therefore, significant efforts have been made to alleviate this impact by pre-processing image sequences to remove the effects of rain and snow. In (Garg and Nayar (2004)), (Ren et al. (2017)), for example using a model based on matrix decomposition to remove the effects of both snow and rain in the latter case. In contrast, (Li et al. (2016)) removes the effects of rain streaks from a single image by learning the static and dynamic background using a Gaussian Mixture Model. A de-noising generator that can remove noise and artefacts induced by the presence of adherent rain droplets and streaks is trained in (Porav et al. (2019)) using data from a stereo rig. A rain mask generated by temporal content alignment of multiple images is also used for keypoint detection (Huang et al. (2019); Yamada et al. (2019)). In spite of these pre-processing strategies, existing visual SLAM and visual odometry (VO) methods tend to be susceptible to these image degradations and tend to perform poorly under such condition.

The quality of LiDAR scans can also be degraded when facing rain droplets, snowflakes and fog particles in extreme weather. A filtering based approach is proposed in (Charron et al. (2018)) to de-noise 3D point cloud scans corrupted by snow before using them for localization and mapping. To mitigate the noisy effects of LiDAR reflection from random rain droplets, (Zhang et al. (2018)) proposes ground-reflectivity and vertical features to build a prior tile map, which is used for localization in a rainy weather. In contrast to process 3D LiDAR scans, (Aldibaja et al. (2016)) suggests the use of 2D LiDAR images reconstructed and smoothed by Principal Component Analysis (PCA). An edge-profile matching algorithm is then used to match the runtime LiDAR images with a mapped set of LiDAR images for localization. However, these methods are not reliable when the rain, snow or fog is moderate or heavy. The results of LIO-SAM (Shan et al. (2020)), a LiDAR based odometry and mapping algorithm fused with IMU data, in light snow show that a LiDAR based approach can work to some degree in snow. However, as the snow increases, the reconstructed 3D point cloud map is corrupted to a high degree with random points from the reflection of snowflakes, which reduces the map’s quality and its re-usability for localization.

In summary, camera and LiDAR sensors are naturally sensitive to rain, fog and snow. Therefore, attempts to use these sensors to perform localization and mapping tasks in adverse weather are limited.

2.2. Radar-based localization and mapping

Using Millimetre Wave (MMW) radar as a guidance sensor for autonomous vehicle navigation can be traced back two or three decades. An Extended Kalman Filter (EKF) based beacon localization system is proposed by (Clark and Durrant-Whyte (1998)) where the wheel encoder information is fused with range and bearing obtained by radar. One of the first substantial solutions for MMW radar-based SLAM is proposed in (Dissanayake et al. (2001)), detecting features and landmarks from radar to provide range and bearing information. (Jose and Adams (2005)) further extends the landmark description and formalizes an augmented state vector containing rich absorption and localization information about targets. A prediction model is formed for the augmented SLAM state. Instead of using the whole radar measurement stream to perform scan matching, (Chandran and Newman (2006)) suggests treating the measurement sequence as a continuous signal and proposes a metric to assess the quality of map and estimate the motion by maximizing the map quality. A consistent map is built using a FMCW radar, an odometer and a gyroscope (Rouveure et al. (2009)). Specifically, vehicle motion is corrected using an odometer and a gyrometer while the map is updated by registering radar scans. Instead of extracting and registering feature points, (Checchin et al. (2010)) uses the Fourier-Mellin Transform (FMT) to estimate the relative transformation between two radar images. In (Vivet et al. (2013)), two approaches are evaluated for localization and mapping in a semi-natural environment using only a radar. The first one is the aforementioned FMT computing relative transformation from whole images, while the second one uses a velocity prior to correct a distorted scan (Vivet et al. (2012)). However, both methods are evaluated without any loop closure detection. A landmark based pose graph radar SLAM system proves that it can work in dynamic environments (Schuster et al. (2016)), (Marck et al. (2013)) use an Iterative Closest Point algorithm (ICP) to register the returned radar point cloud and a Particle filter to map the indoor environment. (Park et al. (2019)) studies the localization to a prior LiDAR map using radar in a low visibility situation. A low-cost millimetre-wave radar is used to provide robust ego-motion estimation in indoor environments in (Almalioglu et al. (2021)) with a RNN-based motion model.

Recently, FMCW radar sensors have been increasingly adopted for vehicles and autonomous robots. (Cen and Newman (2018)) extract meaningful landmarks for robust radar scan matching, demonstrating the potential of using radar to provide odometry information for mobile vehicles in dynamic city environments. This work is extended with a graph based matching algorithm for data association (Cen and Newman (2019)). Radar odometry might fail in challenging environments, such as a road with hedgerows on both sides. Therefore, (Aldera et al. (2019b)) train a classifier to detect failures in the radar odometry using inertial measurements as supervision to automatically label good and bad odometry estimation. Recently, a direct radar odometry method has been proposed to estimate relative pose using FMT, with local graph optimization to further boost the performance (Park et al. (2020)). In (Burnett et al. (2021)), they study the necessity of motion compensation
and Doppler effects on the recent emerging spinning radar for urban navigation.

Deep Learning based radar odometry and localization approaches have been explored in (Barnes et al. (2020b), Aldera et al. (2019a), Barnes and Posner (2020), Gadd et al. (2020), De Martini et al. (2020), Gadd et al. (2021), Tang et al. (2020a,b), Săftescu et al. (2020), Wang et al. (2021)). Specifically, in (Aldera et al. (2019a)) the coherence of multiple measurements is learnt to decide which information should be kept in the readings. In (Barnes et al. (2020b)), a mask is trained to filter out the noise from radar data and Fast Fourier Transform (FFT) cross correlation is applied to the masked images to compute the relative transformation. The experimental results show impressive accuracy of odometry using radar. A self-supervised framework is also proposed for robust keypoint detection on Cartesian radar images which are further used for both motion estimation and loop closure detection (Barnes and Posner (2020)). A hierarchical approach to place recognition and pose refinement for FMCW radar localization is presented in (De Martini et al. (2020)) with compelling performance by using one experience. Promising result for radar based place recognition is shown in (Gadd et al. (2021)) by learning and embedding from sequences in an unsupervised manner.

Full radar-based SLAM systems are able to reduce drift and generate a more consistent map once a loop is closed. A real-time pose graph SLAM system is proposed in (Holder et al. (2019)), which extracts keypoints and computes the GLARE descriptor (Himstedt et al. (2014)) to identify loop closure. However, the system depends on other sensory information, for example, rear wheel speed, yaw rates and steering wheel angles.

2.2.1. Adverse weather. Although radar is considered more robust in adverse weather, the aforementioned methods do not directly demonstrate its operation in these conditions. (Yoneda et al. (2018)) proposes a radar and GNSS/IMU fused localization system by matching query radar images with mapped ones, and tests radar-based localization in three different snow conditions: without snow, partially covered by snow and fully covered by snow. It shows that the localization error grows as the volume of snow increases. However, they did not evaluate their system during snow but only afterwards. To explore the full potential of FMCW radar in all weathers, our previous work (Hong et al. (2020)) proposes a feature matching based radar SLAM system and performs experiments in adverse weather conditions without the aid of other sensors. It demonstrates that radar-based SLAM is capable of operating even in heavy snow when LiDAR and camera both fail. In other interesting recent work, ground penetrating radar is used for localization in inclement weather (Ort et al. (2020)) This takes a completely different perspective to address the problem. The ground penetrating radar (GPR) is utilized for extracting stable features beneath the ground. During the localization stage, the vehicle needs an IMU, a wheel encoder and GPR information to localize.

In this work, we extend our preliminary results presented in (Hong et al. (2020)) with a novel motion estimation algorithm optimally compensating motion distortion and an improved loop closure detection. We also carry out extensive additional experiments, demonstrating more tests and results of Radar SLAM system operating in various weather conditions, including the MulRan dataset and more sequences in adverse weather conditions.

3. Radar sensing and system overview

In this section, we describe the working principle of a FMCW radar and its sensor model. We also elaborate the challenges of employing a FMCW radar for localization and mapping.

3.1. Notation

Throughout this paper, a reference frame $j$ is denoted as $\mathcal{F}_j$ and a homogeneous coordinate of a 2D point $\mathcal{F}_j$ in frame $\mathcal{F}_j$ is defined as $p_j = [x_j, y_j, 1]^T$. A homogeneous transformation $T_{ij} \in \text{SE}(2)$ which transforms a point from the coordinate frame $\mathcal{F}_j$ to $\mathcal{F}_i$ is denoted by a transformation matrix

$$T_{ij} = \begin{bmatrix} R_{ij} & t_{ij} \\ 0 & 1 \end{bmatrix}$$

where $R_{ij} \in \text{SO}(2)$ is the rotation matrix and $t_{ij} \in \mathbb{R}^2$ is the translation vector. Perturbation $\omega \in \mathbb{R}^3$ around the pose $T_{ij}$ uses a minimal representation and its Lie algebra representation is expressed as $\omega \in \mathfrak{se}(2)$. We use the left multiplication convention to define its increment on $T_{ij}$ with an operator $\oplus$, that is

$$\omega \oplus T_{ij} = \exp(\omega') \cdot T_{ij}$$

A polar radar image and its bilinear interpolated Cartesian counterpart are denoted as $\mathbf{S}$ and $\mathbf{I}$, respectively. A point in the Cartesian image $\mathbf{I}$ is represented by its pixel coordinates $\mathbf{p} = [u, v]^T$.

3.2. Geometry of a rotating frequency-modulated continuous-wave radar

There are two types of continuous-wave radar: unmodulated and frequency-modulated radars. Unmodulated continuous-wave radar can only measure the relative velocity of targeted objects using the Doppler effect, while a FMCW radar is also able to measure distances by detecting time shifts and/or frequency shifts between the transmitted and received signals. Some recently developed FMCW radars make use of multiple consecutive observations to calculate targets’ speeds so that Doppler processing is strictly required. This improves the processing performance and accuracy of target range measurements.
Assume a radar sensor rotates 360° clockwise in a full cycle with a total of $N_s$ azimuth angles as shown in Figure 2(a), that is, the step size of the azimuth angle is $2\pi/N_s$. For each azimuth angle, the radar emits a beam and collapses the return signal to the point where a target is sensed along a range without considering elevation. Therefore, a radar image is able to provide absolute metric information of distance, different from a camera image which lacks depth by nature. As shown in Figure 2(b), given a point $(a, r)$ in a polar image $S$ where $a$ and $r$ denote its azimuth and range, its homogeneous coordinates $p$ can be computed by

$$p = \begin{bmatrix} \mu_p \cdot r \cdot \cos \theta \\ \mu_p \cdot r \cdot \sin \theta \\ 1 \end{bmatrix}$$

where $\theta = -a \cdot 2\pi/N_s$ is the ranging angle in Cartesian coordinates, and $\mu_p$ (m/pixel) is the scaling factor between the image space and the world metric space. This point on the polar image can also be related to a point on the Cartesian image $I$ with a pixel coordinate $P$ by

$$u = \frac{w}{2} \cdot \frac{\mu_p}{\mu_c} \cdot r \cdot \sin \theta$$

$$v = \frac{h}{2} \cdot \frac{\mu_p}{\mu_c} \cdot r \cdot \cos \theta$$

where $w$ and $h$ are the width and height of the Cartesian image, and $\mu_c$ (m/pixel) is the scale factor between the pixel space and the world metric space used in the Cartesian image. Therefore, the raw polar scan $S$ can be transformed into a Cartesian space, represented by a grey-scale Cartesian image $I$ through bilinear interpolation, as shown in Figure 2(b).

### 3.3. Challenges of radar sensing for simultaneous localization and mapping

Despite the increasingly widespread adoption of radar systems for perception in autonomous robots and in Advanced Driver-Assistance Systems (ADAS), there are still significant challenges for an effective radar SLAM system.

#### 3.3.1. Coupled artefacts

As a radioactive sensor, radar suffers from multiple sources of artefacts and clutters, for example, speckle noise, receiver saturation and multi-path reflection, as shown in Figure 3(a). Speckle noise is the product of interaction between different radar waves which introduces light and dark random noisy pixels on the image. Meanwhile, multi-path reflection may create ‘ghost’ objects, presenting repetitive similar patterns on the image. The interaction of these multiple sources adds another dimension of complexity and difficulty when applying traditional vision based SLAM techniques to radar sensing.

#### 3.3.2. Discontinuities of detection

Radar operates at a longer wavelength than LiDAR, offering the advantage of perceiving beyond the closest object on a line of sight. However, this could become problematic for some key tasks in pose estimation, for example, frame-to-frame feature matching and tracking, since objects or clutter detected (not detected) in the current radar frame might suddenly disappear (appear) in next frame. As shown in Figure 3(b), this can happen even during a small positional change. This discontinuity of detection can introduce ambiguities and challenges for SLAM, reducing robustness and accuracy of motion estimation and loop closure.

#### 3.3.3. Motion distortion

In contrast to camera and LiDAR, current mechanical scanning radar operates at a relatively low frame rate (4 Hz for our radar sensor). Within a full 360-degree radar scan, a high-speed vehicle can travel several metres and degrees, causing serious motion distortion and discontinuities on radar images, in particular between scans at 0 and 360°. An example in Figure 3(c) shows this issue on the Cartesian image on the left, that is, skewed radar detections due to motion distortion. By contrast, there are no skewed detections when it is static. Therefore, directly using these distorted Cartesian images for geometry estimation and mapping can introduce errors.
3.4. System overview

Having these challenges in mind, we propose a novel radar SLAM system which includes motion compensated radar motion estimation, loop closure detection and pose graph optimization. The system, shown in Figure 4, is divided into two threads. The main thread is the tracking thread which takes the Cartesian images as input, tracks the radar motion and creates new points and keyframes for mapping. The other parallel thread takes the polar images as input and is responsible for generation of the dense point cloud and computation of descriptors for loop closure detection. Finally, once a loop is detected, it performs pose graph optimization to correct the drift induced by tracking before updating the map.

4. Radar motion estimation

This section describes the proposed radar motion estimation algorithm, which includes feature detection and tracking,
graph based outlier rejection and radar pose tracking with optimal motion distortion compensation.

4.1. Feature detection and tracking

For each radar Cartesian image $I_j$, we first detect keypoints purely using a blob detector based on a Hessian matrix. Keypoints with Hessian responses larger than a threshold are selected as candidate points. The candidate points are then selected based on the adaptive non-maximal suppression (ANMS) algorithm (Bailo et al. (2018)), which selects points that are homogeneously spatially distributed. Instead of using a descriptor to match keypoints as in (Hong et al. (2020)), we track them between frames $I_j$ and $I_{j-1}$ using the KLT tracker (Lucas and Kanade (1981)).

4.2. Graph based outlier rejection

It is inevitable that some keypoints are detected and tracked on dynamic objects, for example, cars, cyclists and pedestrians, and on radar noise, for example, multi-path reflection. We leverage the absolute metrics that radar images directly provide to form geometric constraints used for detecting and removing these outliers.

We apply a graph based outlier rejection algorithm described in (Howard (2008)). We impose a pairwise geometric consistency constraint on the tracked keypoint pair based on the fact that they should follow a similar motion tendency. The assumption is that most of the tracked points are from static scene data. Therefore, for any two pairs of keypoint matches between the current $I_j$ and the last $I_{j-1}$ radar frames, they should satisfy the following pairwise constraint:

$$\left\| P_{m_{j-1}} - P_{m_j} \right\|_2 - \left\| P_{n_{j-1}} - P_{n_j} \right\|_2 < \delta_c$$

where $\cdot$ is the absolute operation, $\left\| \cdot \right\|_2$ is the Euclidean distance, $\delta_c$ is a small distance threshold, and $P_{m_{j-1}}$, $P_{n_{j-1}}$, $P_{m_j}$ and $P_{n_j}$ are the pixel coordinates of two pairs of tracked points between $I_{j-1}$ and $I_j$. Hence, $P_{m_{j-1}}$ and $P_{n_{j-1}}$ denote a pair of associated points while $P_{m_j}$ and $P_{n_j}$ is another pair, see Figure 5 for an intuitive example. A consistency matrix $G$ is then used to represent all the associations that satisfy this constraint. If a pair of associations satisfies this constraint, the corresponding entry in $G$ is set as one shown in Figure 5. Finding the maximum inlier set of all matches that are mutually consistent is equivalent to deriving the maximum clique of a graph represented by $G$, which can be solved efficiently using (Konc and Janezic (2007)). Once the maximum inlier set is obtained, it is used to compute the relative transformation $T_{j-1,j}$ which transforms a point from local frame $j$ to local frame $j-1$ using Singular Value Decomposition (SVD) (Challis (1995)). Given $T_{j-1,j}$ and the fixed radar frame rate, an initial guess of current velocity $v_j = [v_x, v_y, v_\theta]^T \in \mathbb{R}^3$ can be computed for the motion compensation tracking model. Similarly, a graph based method is applied to perform outlier rejection in (Cen and Newman (2018, 2019)). Global geometric constraints are imposed in both methods. However, their method formulates the graph matching problem as an optimization problem maximizing the global compatibility, while ours solves the maximum clique problem with an approximate vertex-colouring algorithm.

4.3. Motion distortion modelling

After the tracked points are associated, they can be used to estimate the motion. However, since the radar scanning rate...
is slow, they tend to suffer from serious motion distortion as discussed in Sec. 3.3. This can dramatically degrade the accuracy of motion estimation, which is different from most of the vision and LiDAR based methods. Therefore, we explicitly model and compensate for motion distortion in radar pose tracking using an optimization approach.

Assume a full polar radar scan $S_j$ takes $\Delta t$ seconds to finish. Denote $T_{wj}$ as the pose of radar scan $S_j$ in the world coordinate frame $F_w$ and $T_{j,b}$ as the pose of the radar scan in the local frame while capturing its azimuth beam at time $t \in [-\Delta t/2, \Delta t/2]$. Without losing generality, we compensate the motion distortion relative to the central azimuth beam at $t = 0$, that is, $F_b$ defines the local coordinate frame $F_j$ of $S_j$. The motion distortion model is designed to correct detections on each beam of a radar scan, that is, optimally estimating the detections on an undistorted radar image as shown in Figure 6.

The radar pose in the world coordinate frame while capturing an azimuth scan at time $t$ can be obtained by

$$T_{wb} = T_{wj} T_{jb}$$

(6)

Consider a constant velocity model in a full scan, we can compute the relative transformation $T_{jb}$ given the velocity $v_j$, that is

$$T_{jb} = \exp\left(\left(v_j t\right)^T\right) = \begin{bmatrix} \cos(v_j t) & -\sin(v_j t) & v_j t \\ \sin(v_j t) & \cos(v_j t) & v_j t \\ 0 & 0 & 1 \end{bmatrix}$$

(7)

where $\exp()$ is the matrix exponential map and $\bigwedge$ is the operation to transform a vector to a matrix. If the $i$th keypoint $p_i^j$ in the world frame $F_w$ is observed as $p_i^j$ in the azimuth scan at time $t$, its motion compensated location is then

$$p_i^{jb} = T_{jb} p_i^j$$

(8)

In other words, $p_i^{jb}$ is the compensated location of $p_i^j$ in the local frame $F_b$. Therefore, the feature residual between the locally observed and estimated (after motion compensation) locations of this $i$th keypoint can be computed as:

$$e_i^j = \rho_c \left(T_{wj} p_i^w - p_i^{jb}\right)$$

(9)

where $\rho_c$ is the Cauchy robust cost function used to account for perspective changes described in 3.3.2.

### 4.4. Optimal motion compensated radar pose tracking

Radar pose tracking aims to find the optimal radar pose $T_{wj}$ and the current velocity $v_j$ while considering the motion distortion. In order to ensure smooth motion dynamics, a velocity error $e_v$ is also introduced as a velocity prior term:

$$e_v = v_j - v_{jprior}$$

(10)

where $v_{jprior}$ is a prior on the current velocity which is parameterized as

$$v_{jprior} = \log\left((T_{wj\rightarrow}^{-1} T_{wj})^V\right) / \Delta t$$

(11)

Here, $\bigwedge$ is the operation to convert a matrix to a vector. This velocity prior term establishes a constraint on velocity changes by considering the previous pose $T_{wj\rightarrow\rightarrow}$. This prior is crucial to stabilize the optimization. The results with and without this prior are compared in Figure 7. Therefore, the pose tracking optimizes the velocity $v_j$ and the current radar pose $T_{wj}$ by minimizing the cost function including the feature residuals of all the $N$ keypoints tracked in $S_j$ and the velocity prior, that is

$$v_j, T_{wj} = \arg \min_{v_j, T_{wj}} \left\{ e_v^T \Lambda_v e_v + \sum_{i=1}^N e_i^j \Lambda_i^j e_i^j \right\}$$

(12)

where $\Lambda_v^j$ and $\Lambda_i^j$ are the information matrices of the keypoint $i$ and the velocity.

By formulating a state variable $\Theta$ containing all the variables to be optimized, that is, the velocity $v_j$ and the current radar pose $T_{wj}$, and denoting $e(\Theta)$ as the residual function containing both $e_v$ and $e_i$, $\Theta$ can be solved in an optimization whose factor graph representation is shown in Figure 8. The optimization problem is then framed as finding the minimum of the weighted Sum of Squared Errors cost function:

$$F(\Theta) = \sum_i e_i^j(\Theta)^T W_i e_i(\Theta)$$

(13)

$$\Theta^* = \arg \min_{\Theta} F(\Theta)$$

(14)

where $W_i$ is a symmetric positive definite weighting matrix.

The total cost in 13 is minimized using the Levenberg-Marquardt algorithm. With the initial guess $\Theta$, the residual
$e_i(\Delta\Theta \oplus \Theta)$ is approximated by its first order Taylor expansion around the current guess $\Theta$:

$$e_i(\Delta\Theta \oplus \Theta) = e_i + J_i \Delta\Theta \quad (15)$$

where $J_i$ is the Jacobian of $e_i$ evaluated with the current parameters $\Theta$. $\oplus$ represents a standard addition operator for the velocity variable $v_j$ and a $\wedge$ operation for the pose variable $T_{w,j}$, that is, $\exp ((\Delta\Theta)^\wedge) \cdot T_{w,j}$ (Kümmerle et al. (2011), Schubert et al. (2018)). $J_i$ can be decomposed into two parts with respect to the velocity and the pose

$$J_i = [J_v, J_p] \quad (16)$$

that is, the Jacobian part with respect to the velocity as

$$J_v = \frac{\partial e_i(\Theta)}{\partial (v_j t)} \quad (17)$$

and the Jacobian with respect to the pose as

$$J_p = \frac{\partial e_i(\Theta)}{\partial (T_{w,j})} \quad (18)$$

which is equivalent to computing the Jacobian with respect to zero perturbation $\Delta\Theta_p$ around the pose (Solà et al. (2018)), that is

$$\frac{\partial e_i(\Theta)}{\partial (T_{w,j})} = \frac{\partial e_i(\Delta\Theta \oplus \Theta)}{\partial \Delta\Theta_p} \quad (19)$$

The Levenberg-Marquardt algorithm computes a solution $\Delta\Theta$ at each iteration such that it minimizes the residual function $e_i(\Delta\Theta \oplus \Theta)$. We refer the reader to (Kümmerle et al. (2011)) for further details on the optimization process.

### 4.5. New point generation

After tracking the current radar scan $S_j$, the total number of successfully tracked keypoints is checked to decide whether new keypoints should be generated. If it is below a certain threshold, new keypoints are extracted and added for tracking if they are located in image grids whose total numbers of keypoints are low. Specifically, we divide the radar image into grids and count the number of tracked points in each grid. These grids are then sorted according to the number of keypoints in each of them. For the grids that have fewer number of keypoints, new keypoints are extracted in them before being inserted for tracking. The number of the new keypoints is limited by the maximum number of keypoints in each grid. Once a new keypoint $p_{ij}^w$ is associated with the current frame $S_j$, its global position $p_{ij}^w$ can be derived from

$$p_{ij}^w = T_{w,j}^w \cdot \exp(v_j^t) \cdot p_{ij} \quad (20)$$

using its direct observation of $p_{ij}^w$ (with distortion). $T_{w,j}^w$ and $v_j^t$ are the optimal radar pose and velocity derived in equation (12).

### 4.6. New keyframe generation

To scale the system in a large-scale environment, we use a pose-graph representation for the map with each node parameterized by a keyframe. Each keyframe which contains a velocity and a pose is connected with its neighbouring keyframes using its odometry derived from the motion estimation. The keyframe generation criterion is similar to that introduced in (Mur-Artal and Tardós (2017)) based on travelled distance and angle.

### 5. Loop closure and pose graph optimization

Robust loop closure detection is critical to reduce drift in a SLAM system. Although the Bag-of-Words model has proved efficient for visual SLAM algorithms, it is not adequate for radar-based loop closure detection due to three main reasons: first, radar images have less distinctive pixel-wise characteristics compared to optical images, which means...
similar feature descriptors can repeat widely across radar images causing a large number of incorrect feature matches; second, the multi-path reflection problem in radar can introduce further ambiguity for feature description and matching; third, a small rotation of the radar sensor may produce tremendous scene changes, significantly distorting the histogram distribution of the descriptors. There are some attempts to address this challenge in the context of place recognition (Săftescu et al. (2020), Gadd et al. (2021), Kim et al. (2020)). On the other hand, radar imagery encapsulates valuable absolute metric information, which is inherently missing for an optical image. Therefore, we propose a loop closure technique which captures the geometric scene structure and exploits the spatial signature of reflection density from radar point clouds.

Algorithm 1. Radar Polar Scan to Point Cloud Conversion

Input: Radar polar scan \( S \in \mathbb{R}^{m \times n} \);
Output: Point Cloud \( C \in \mathbb{R}^{c \times 2} \);
Parameters: Minimum peak prominence \( \delta_p \) and minimum peak distance \( \delta_d \)

1. Initialize empty point cloud set \( C \);
2. for \( i \leftarrow 1 \) to \( m \) do
3. \( Q^{k+1} \leftarrow \text{findPeaks} (S[i, :) \), \( \delta_p \), \( \delta_d \); \)
4. \( (\mu, \sigma) \leftarrow \text{meanAndStandardDeviation} (Q^{k+1}); \)
5. for each peak \( q \) in \( Q \) do
6. if \( q \geq (\mu + \sigma) \) then
7. \( p \leftarrow \text{transformPeakToPoint} (q, i); \)
8. Add the point \( p \) to \( C \);
9. end
10. end

5.1. Point cloud generation

Considering the challenges of radar sensing in Sec. 3.3, we want to separate true targets from the noisy measurements on a polar scan. An intuitive and naive way would be to detect peaks by finding the local maxima from each azimuth reading. However, as shown in Figure 9, the detected peaks can be distributed randomly across the whole radar image, even for areas without a real object, due to the speckle noise described in Sec. 3.3.1. Therefore, we propose a simple yet effective point cloud generation algorithm using adaptive thresholding. We denote the return power of a peak as \( q \), we select peaks which satisfy the following inequality constraint

\[ q \geq \mu + \sigma \]  

where \( \mu \) and \( \sigma \) are the mean and the standard deviation of the powers of the peaks in one azimuth scan. Estimating the power mean and standard deviation along one azimuth instead of the whole polar image can mitigate the effect of receiver saturation since the radar may be saturated at one direction while rotating. By selecting the peaks whose powers are greater than one standard deviation plus their mean power, the true detections tend to be separated from the false-positive ones. The procedure is shown in Algorithm 1. Once a point cloud \( C \) is generated from a radar image, M2DP (He et al. (2016)), a rotation invariant global descriptor designed for 3D point clouds, is adapted for the 2D radar point cloud to describe it for loop closure detection. M2DP computes the density signature of the point cloud on a plane and uses the left and right singular vectors of these signatures as the descriptor.

5.2. Loop Candidate Rejection with principal component analysis

We leverage principal component analysis (PCA) to determine whether the current frame can be a candidate to be matched with historical keyframes for loop closure detection. After performing PCA on the extracted 2D point cloud \( C \), we compute the ratio \( r_{pca} = \gamma_1/\gamma_2 \) between the two eigenvalues \( \gamma_1 \) and \( \gamma_2 \) where \( \gamma_1 \geq \gamma_2 \). The frame is selected for loop closure detection if its \( r_{pca} \) is less than a certain threshold \( \delta_{pca} \). The intuition behind this is to detect loop closure mainly on point clouds which have distinctive structural layouts, reducing the possibility of detecting false-positive loop closures. In other words, if \( \gamma_1 \) is dominant (that is, \( r_{pca} \) is big), it is very likely that the radar imagery is collected in an environment, such as a highway or a country road, which exhibits less distinctive structural patterns and layouts and should be avoided for loop closure detection. Some examples are given in Figure 10.

Figure 9. Peak detection in a radar scan. Left: Original Cartesian image. Middle: Peaks (in yellow) detected using a local maxima algorithm. Note that a great amount of peaks detected are due to speckle noise. Right: Peaks detected using the proposed point cloud extraction algorithm which preserves the environmental structure and suppresses detections from multi-path reflection and speckle noise.
5.3. Relative transformation

Once a loop closure is detected, the relative transformation $T_{lj}$ between the current radar image $I_j$ and the matched radar image $I_l$ is computed. Similar to Sec. 4.1, we also associate keypoints of the two frames by using KLT tracker. The challenge here is that $I_l$ might have a large rotation with respect to $I_j$, causing the tracker to fail. To address this problem, we estimate firstly the relative rotation between the two frames and align them by using the eigenvectors from the PCA of their point clouds, similar to Sec. 5.2. Then, the keypoints of $I_l$ are tracked through the rotated version of $I_j$. After obtaining the keypoint association, ICP is used to compute the relative transformation $T_{lj}$, which is added in the pose graph as a loop closure constraint for pose graph optimization.

5.4. Pose graph optimization

A pose graph is gradually built as the radar moves. Once a new loop closure constraint is added in the pose graph, pose graph optimization is performed. After successfully optimizing the poses of the keyframes, we update the global map points. The g2o (Kümmerle et al. (2011)) library is used in this work for the pose graph optimization.

6. Experimental results

Both quantitative and qualitative experiments are conducted to evaluate the performance of the proposed radar SLAM method using three open radar datasets, covering large-scale environments and some adverse weather conditions.

6.1. Evaluation protocol

We perform both quantitative and qualitative evaluation using different datasets. Specifically, the quantitative evaluation is to understand the pose estimation accuracy of the SLAM system. For Relative/Odometry Error (RE), we follow the popular KITTI odometry evaluation criteria, that is, computing the mean translation and rotation errors from length 100–800 m with a 100 m increment. Absolute Trajectory Error (ATE) is also adopted to evaluate the localization accuracy of full SLAM, in particular after loop closure and global graph optimization. The trajectories of all methods (see full list in Sec. 6.3) are aligned with the ground truth trajectories using a 6 Degree-of-Freedom (DoF) transformation provided by the evaluation tool in (Zhang and Scaramuzza (2018)) for ATE evaluation. On the other hand, the qualitative evaluation focuses on how some challenging scenarios, for example, in adverse weather conditions, influence the performance of various vision, LiDAR and radar-based SLAM systems.

6.2. Datasets

So far there exist three public datasets that provide long-range radar data with dense returns: the Oxford Radar RobotCar Dataset (Barnes et al. (2020a), Maddern et al. (2017)), the MulRan Dataset (Kim et al. (2020)) and the RADIATE Dataset (Sheeny et al. (2021)). We choose the Oxford RobotCar and MulRan datasets for detailed quantitative benchmarking and our RADIATE dataset mainly for qualitative evaluation in our experiments.

6.2.1. Oxford radar RobotCar dataset. The Oxford Radar RobotCar Dataset (Barnes et al. (2020a), Maddern et al. (2017)) provides data from a Navtech CTS350-X Millimetre-Wave W radar for about 280 km of driving in Oxford, UK, traversing the same route 32 times. It also provides stereo images from a Point Grey Bumblebee XB3 camera and LiDAR data from two Velodyne HDL-32E sensors with ground truth pose locations. The radar is configured to provide 4.38 cm and 0.9° resolution in range and azimuth, respectively, with a range up to 163 m. The radar scanning frequency is 4 Hz. See Figure 11 for some examples of data.

6.2.2. MulRun dataset. The MulRun Dataset (Kim et al. (2020)) provides radar and LiDAR range data, covering multiple cities at different times in a variety of city environments (e.g. bridge, tunnel and overpass). A Navtech CIR204-H Millimetre-Wave FMCW radar is used to obtain radar images with 6 cm range and 0.9° rotation resolutions with a maximum range of 200 m. The radar scanning frequency is also 4 Hz. It also has an Ouster 64-channel LiDAR sensor operating at 10 Hz with a maximum range of 120 m. Different routes are selected for our experiments, including Dajeon Convention Center (DCC), KAIST and Riverside. Specifically, DCC presents diverse structures while KAIST is collected while moving within a campus. Riverside is captured along a river and two bridges with repetitive features. Each route contains three traverses on different days. Some LiDAR and radar data examples are given in Figure 12.
6.2.3. RADIATE dataset. The RADIATE dataset is our recently released dataset which includes radar, LiDAR, stereo camera and GPS/IMU data (Sheeny et al. (2021)). One of its unique features is that it provides data in extreme weather conditions, such as rain and snow, as shown Figure 13. A Navtech CIR104-X radar is used with 0.175 m range resolution and maximum range of 100 m at 4 Hz operating frequency. A 32-channel Velodyne HDL-32E LiDAR and a ZED stereo camera are set at 10 Hz and 15 Hz, respectively. The seven sequences used in this work include 2 fog, 1 rain, 1 normal, 2 snow and 1 night recorded in the City of Edinburgh, UK. Their sequence lengths are given in Table 1. Note that only the rain, normal, snow and night sequences have loop closures and the GPS signal is occasionally lost in the snow sequence.

6.3. Competing methods and their settings

In order to validate the performance of our proposed radar SLAM system, state-of-the-art odometry and SLAM methods for large-scale environments using different sensor modalities (camera, LiDAR, radar) are chosen. These include ORB-SLAM2 (Mur-Artal and Tardós (2017)), SuMa (Behley and Stachniss (2018)) and our previous version of RadarSLAM (Hong et al. (2020)), as baseline algorithms for vision, LiDAR and radar-based approaches, respectively.
For the Oxford Radar RobotCar Dataset, the results reported in (Cen and Newman (2018), Barnes et al. (2020b)) are also included as a radar-based method due to the unavailability of their implementations.

We would like to highlight that we use an identical set of parameters for our radar odometry and SLAM algorithm across all the experiments and datasets. We believe this is worthwhile to tackle the challenge that most existing odometry or SLAM algorithms require some levels of parameter tuning in order to reduce or avoid result degradation.

6.3.1. Stereo vision based ORB-SLAM2. ORB-SLAM2 (Mur-Artal and Tardós (2017)) is a sparse feature based visual SLAM system which relies on ORB features. It also possesses loop closure and pose graph optimization capabilities. Local Bundle Adjustment is used to refine the map point position which boosts the odometry accuracy. Based on its official open-source implementation, we use its stereo setting in all experiments and loop closure is enabled.

6.3.2. LiDAR based SuMa. SuMa (Behley and Stachniss (2018)) is one of the state-of-the-art LiDAR based odometry and mapping algorithms for large-scale outdoor environments, especially for mobile vehicles. It constructs and uses a surfel-based map to perform robust data association for loop closure detection and verification. We employ its open-source implementation and keep the original parameter setting used for KITTI dataset in our experiments.

6.3.3. Radar-based radarSLAM. Our previous version of RadarSLAM (Hong et al. (2020)) extracts SURF features from Cartesian radar images and matches the keypoints based on their descriptors for pose estimation, which is different from the feature tracking technique in this work. It does not consider motion distortion although it includes loop closure detection and pose graph optimization to reduce drift and improve the map consistency. It is named as ‘Baseline Odometry’ and ‘Baseline SLAM’ for comparison.

6.3.4. Cen’s radar odometry. Cen’s method (Cen and Newman (2018)) is one of the first attempts using the Navtech FMCW radar sensor to estimate ego-motion of a mobile vehicle. Landmarks are extracted from polar scans before performing data association by maximizing the overall compatibility with pairwise constraints. Given the associated pairs, SVD is used to find the relative transformation.

6.3.5. Barnes’ radar odometry. Barnes’ method (Barnes et al. (2020b)) leverages deep learning to generate distraction-free feature maps and uses FFT cross correlation to find relative poses on consecutive feature maps. After being trained end-to-end, the system is able to mask out multi-path reflection, speckle noise and dynamic objects. This facilitates the cross correlation stage and produces accurate odometry. The spatial cross-validation results in appendix of (Barnes et al. (2020b)) are chosen for fair comparison.

---

**Figure 13.** Images collected in Snow (left), Fog/Rain (middle), and Night (right). The image quality degrades in these conditions, making it extremely challenging for vision based odometry and SLAM algorithms. Note that for the snow sequence, the camera is completely covered by snow.

**Table 1.** Lengths of collected sequences in RADIATE Dataset.

| Sequence  | Fog 1 | Fog 2 | Rain | Normal | Snow 1 | Snow 2 | Night |
|-----------|-------|-------|------|--------|--------|--------|-------|
| Length (km) | 4.7   | 4.8   | 3.3  | 3.3    | 8.7    | 3.3    | 5.6   |
6.4. Experiments on RobotCar dataset

Results of eight sequences of RobotCar Dataset are reported here for evaluation, that is, 10-11-46-21, 10-12-32-52, 11-14-02-26, 16-11-53-11, 17-13-26-39, 18-14-14-42, 18-14-46-59 and 18-15-20-12. The wide baseline stereo images are used for the stereo ORB-SLAM2 and the left Velodyne HDL-32E sensor is used for SuMa.

6.4.1. Quantitative comparison. The RE and ATE results of each sequence are given in Table 2 and Table 3, respectively. Since the ground truth poses provided by the Oxford Radar RobotCar Dataset are 3-DoF, only the $x$, $y$ and $\theta$ of the estimated 6-DoF poses of ORB-SLAM2 and SuMa are evaluated. Note that SuMa fails on these eight sequences at $10-30\%$ of the full lengths and its results are therefore reported until the point where it fails, and ATE is not applicable due to the lack of fully estimated trajectories. Specifically, the stereo version of ORB-SLAM2 is able to complete seven sequences, successfully close the loops and achieve superior localization accuracy. SuMa also performs accurately when it works although it is less robust on these sequences. This may be due to the large number of dynamic objects, for example, surrounding moving cars and buses. Regarding radar-based approaches, we can see that our proposed radar odometry/SLAM achieves less RE compared to the baseline radar odometry/SLAM and Cen’s method and a similar mean AE to the learning based Barnes’ method. It can also be seen that our proposed radar odometry and SLAM methods achieve better or comparable RE and ATE performance to ORB-SLAM2 and SuMa.

Figure 14 describes the REs of ORB-SLAM2, SuMa, baseline radar odometry and our radar odometry/SLAM algorithms using different path lengths and speeds, following the popular KITTI odometry evaluation protocol. SuMa has the lowest error on both translation and rotation against path lengths and speed until it fails, while our radar SLAM and odometry methods are the second and third lowest, respectively. The low, median and high translation and rotation errors are presented in Figures 15(a) and (b). It can be seen that our SLAM and odometry achieve low values for both translation and rotation errors for different path lengths. The optimized velocities of $x$, $y$ and $\theta$ are given in Figure 16 compared to the ground truth. The optimized velocities have very high accuracy, which verifies the superior performance of our proposed radar motion estimation algorithm.

6.4.2. Qualitative comparison. We show the estimated trajectories of six sequences in Figure 17 for qualitative evaluation. For most of the sequences, our SLAM results are closest to the ground truth although the trajectories of baseline SLAM and ORB-SLAM2 are also accurate except for sequence 18-14-14-42. Figure 18 elaborate on the trajectory of each method on sequence 17-13-26-39 for qualitative performance.

We further compare the proposed radar odometry with the baseline radar odometry (Hong et al. (2020)). Estimated
trajectories of three sequences are presented in Figure 19. It is clear that our radar odometry drifts much slower than the baseline radar odometry method, validating the superior performance of the motion estimation algorithm with feature tracking and motion compensation. Therefore, our SLAM system also benefits from this improved accuracy.

6.5. Experiments on MulRun dataset

The RE and ATE of SuMa, baseline radar odometry/SLAM and our radar SLAM are shown in Table 4 and Table 5. ORB-SLAM2 is not applicable here since MulRan only contains radar and LiDAR data. Similar to the RobotCat
Figure 16. Optimized x, y and yaw velocities on RobotCar.

Figure 17. Trajectories for six sequences using different SLAM algorithms on Oxford Radar RobotCar Dataset.
dataset, we again transform its provided 6-DoF ground truth poses into 3-DoF for evaluation. Both RE and ATE are evaluated on nine sequences: DCC01, DCC02, DCC03, KAIST01, KAIST02, KAIST03, Riverside01, Riverside02 and Riverside03. In terms of RE, both our odometry and SLAM system achieve comparable or better performance on all sequences.

Our odometry method reduces both translation error and rotation errors significantly compared to the baseline. Our SLAM system, to a great extent, outperforms both baseline SLAM and SuMa on ATE. More importantly, only our SLAM reliably works on all nine sequences which cover diverse urban environments. Specifically, the baseline SLAM detects wrong loop closures on sequence KAIST03, Riverside02 and Riverside03 and fails to detect a loop in DCC03, which causes its large ATEs for these sequences. SuMa, on the other hand, fails to finish the sequences Riverside01, 02 and 03, likely due to the impact of fog on LiDAR sensing.

The estimated trajectories on sequences DCC01, DCC02, DCC03, KAIST01, KAIST02, KAIST03 are shown in Figure 21. These qualitative results of the algorithms provide similar observations to the RE and ATE. For clarity, Figure 22 presents trajectories of the SLAM algorithms on Riverside01 in separate figures.

6.6. Experiments on the RADIATE dataset

To further verify the superiority of radar against LiDAR and camera in adverse weathers and degraded visual environments, we perform qualitative evaluation by comparing the estimated trajectories with a high-precision Inertial Navigation System (inertial system fused with GPS) using our RADIATE dataset. Since ORB-SLAM2 fails to produce meaningful results due to the visual degradation caused by water drops, blurry effects in low-light conditions and occlusion from snow (see Figure 13 for example images), its results are not reported in this section.

6.6.1. Experiments in adverse weather

Estimated trajectories of SuMa, baseline and our odometry for Fog 1 and Fog 2 are shown in Figures 23(a) and 23(b), respectively. We can see that our SLAM drifts less than SuMa and the baseline radar SLAM although they all suffer from drift without loop closure. SuMa also loses tracking for sequence Fog 2, which is likely due to the impact of fog on LiDAR sensing.

The impact of snowflakes on LiDAR reflection is more obvious. Figure 24 shows the LiDAR point clouds of two of the same places in snowy and normal conditions. Depending on the snow density, we can see two types of degeneration of LiDAR in snow. It is clear that both the number of correct LiDAR reflections and point intensity dramatically drop in snow for place 1, while there are a lot of
noisy detections around the origin for place 2. Both cases can be challenging for LiDAR based odometry/SLAM methods. This matches the results of the Snow sequence in Figure 25. Specifically, when the snow was initially light, SuMa was operating well. However, when the snow gradually became heavier, the LiDAR data degraded and eventually SuMa lost track. The three examples of LiDAR scans at the point when SuMa fails are shown in Figure 25.

The very limited surrounding structures sensed by LiDAR makes it extremely challenging for LiDAR odometry/SLAM methods like SuMa. In contrast, our radar SLAM method is still able to operate accurately in heavy snow, estimating a more accurate trajectory than the baseline SLAM.

6.6.2. Experiments on the same route in different weathers.

To compare different algorithms’ performance on the same route but in different weather conditions, we also provide results here in normal weather, rain and snow conditions, respectively. The estimated trajectories of SuMa, baseline SLAM and our SLAM result in normal weather are shown in Figure 26(a) while for the Rain sequence these are shown in Figure 26(b). In the Rain sequence, there is moderate rain. LiDAR based SuMa is slightly affected, and as we can see at the beginning of the sequence, SuMa estimates a shorter length. Our radar SLAM estimates a more accurate trajectory than the baseline SLAM.

In the Snow 2 sequence, there is moderate snow, and the results are shown in Figure 26(c). The Snow 2 sequence was taken while moving quickly. Therefore, without motion the

Table 4. Relative error on MulRan Dataset.

| Sequence | Method | DCC01 | DCC02 | DCC03 | KAIST01 | KAIST02 | KAIST03 | Riverside01 | Riverside02 | Riverside03 | Mean |
|----------|--------|-------|-------|-------|---------|---------|---------|-------------|-------------|-------------|------|
| SuMa     | DCC01  | 2.71/0.4 | 4.07/0.9 | 2.14/0.6 | 2.9/0.8 | 2.64/0.6 | 2.17/0.6 | 1.66/0.6*30% | 1.49/0.5*23% | 1.65/0.4*5% | 2.38/0.5 |
| Baseline Odometry | DCC02  | 3.35/0.9 | 2.12/0.6 | 1.74/0.6 | 2.32/0.8 | 2.69/1.0 | 2.62/0.8 | 2.70/0.7 | 3.09/1.1 | 2.71/0.7 | 2.59/0.8 |
| Baseline SLAM | DCC03  | 3.81/0.9 | 2.04/0.5 | 1.90/5.5 | 2.34/0.7 | 1.95/0.6 | 20.1/5.1 | 3.56/0.9 | 3.05/6.8 | 152/0.175 | 21.1/2.3 |
| Our Odometry | KAIST01 | 2.70/0.5 | 1.90/0.4 | 1.64/0.4 | 2.13/0.7 | 2.07/0.6 | 1.99/0.5 | 2.04/0.5 | 1.51/0.5 | 1.71/0.5 | 1.97/0.5 |
| Our SLAM | KAIST02 | 2.39/0.4 | 1.90/0.4 | 1.56/0.2 | 1.75/0.5 | 1.76/0.4 | 1.72/0.4 | 3.40/0.9 | 1.79/0.3 | 1.95/0.5 | 2.02/0.4 |
|          | KAIST03 |        |       |       |         |         |         |             |             |             |      |
|          | Riverside01 |        |       |       |         |         |         |             |             |             |      |
|          | Riverside02 |        |       |       |         |         |         |             |             |             |      |
|          | Riverside03 |        |       |       |         |         |         |             |             |             |      |

Results are given as translation error/rotation error. Translation error is in %, and rotation error is in degrees per 100 m (deg/100 m). * indicates the algorithm fails at the xx% of the sequence and its result is reported up to that point.

Table 5. Absolute trajectory error for position (RMSE) on MulRan Dataset.

| Sequence | Method | DCC01 | DCC02 | DCC03 | KAIST01 | KAIST02 | KAIST03 | Riverside01 | Riverside02 | Riverside03 |
|----------|--------|-------|-------|-------|---------|---------|---------|-------------|-------------|-------------|
| SuMa     | DCC01  | 13.509 | 17.834 | 29.574 | 38.693 | 31.864 | 45.970 | N/A         | N/A         | N/A         |
| Baseline SLAM | DCC02  | 17.458 | 24.962 | 76.138 | 4.931  | 3.918  | 50.809 | 10.531      | 1091.605    | 95.247      |
| Our SLAM | DCC03  | 12.886 | 9.878 | 3.917 | 6.873 | 6.028 | 4.109 | 9.029 | 7.049 | 10.741 |

The absolute trajectory error of position is in metres. N/A: SuMa fails to finish Riverside01, 02 and 03 sequences.

Figure 20. Trajectories results of different SLAM algorithms on sequence Riverside01 of MulRan.

Figure 21. Riverside scenery of MulRan from Google Street View: repetitive structures are challenging to LiDAR based methods moving on high way.
compensation, the baseline SLAM drifts heavily and cannot close the loop while our SLAM consistently performs well. Hence, the results in Figure 26 once again confirm that our proposed SLAM system is robust in all weather conditions.

6.6.3. Experiments at night. The estimated trajectories of SuMa, baseline SLAM and our SLAM on the Night sequence are shown in Figure 23(c). LiDAR based SuMa is almost unaffected by the dark night although it does not detect the loops. Both baseline and our SLAM perform well in the night sequence, producing more accurate trajectories after detecting loop closures.

6.7. Average completion percentage

We calculate the average completion percentage for each competing algorithm on each dataset, to evaluate the robustness of each algorithm representing a different sensor modality. The number of frames that a method completed before losing tracking is denoted as $K_{\text{completed}}$ while the total number of frames is denoted as $K_{\text{total}}$. The metric is computed as:

$$\text{Percentage} = \frac{K_{\text{completed}}}{K_{\text{total}}} \times 100\%$$  \hspace{1cm} (22)

The MulRan dataset does not include camera data so it is shown as N/A for ORB-SLAM2. In the RADIATE dataset, the camera is either blocked by snow or blurred in the night so ORB-SLAM2 fails to initialize and it is also shown as N/A. In Table 6 we can see that only the radar-based methods are reliable and completed in all cases. Neither vision based nor LiDAR based methods manage to finish on all three datasets.
6.8. Parameters used

The same set of parameters provided in Table 7 is employed in all the experiments, covering different cities, radar resolutions and ranges, weather conditions, road scenarios, etc. This selected set of parameters is tuned using the 16-13-09-37 sequence in the Oxford Radar RobotCar Dataset.

6.9. Runtime

The system is implemented in C++ without a GPU. The computation time of a tracking thread is shown in Figure 27 showing that our proposed system runs at 8 Hz, which is twice as fast as the 4 Hz radar frame rate, on a laptop with an Intel i7 2.60 GHz CPU and 16 GB RAM. The loop closure and pose...
graph optimization are performed with an independent thread which does not affect our real-time performance.

7. Conclusion
In this paper, we have presented a FMCW radar-based SLAM system that includes pose tracking, loop closure and pose graph optimization. To address the motion distortion problem in radar sensing, we formulate pose tracking as an optimization problem that explicitly compensates for the motion without the aid of other sensors. A robust loop closure detection scheme is specifically designed for the FMCW radar. The proposed system is agnostic to the radar resolutions, radar range, environment and weather conditions. The same set of system parameters is used for the evaluation of three different datasets covering different cities and weather conditions.

Extensive experiments show that the proposed FMCW radar SLAM algorithm achieves comparable localization accuracy in normal weather compared to the state-of-the-art LiDAR and vision based SLAM algorithms. More importantly, it is the only one that is resilient to adverse weather conditions, for example, snow and fog, demonstrating the superiority and promising potential of using FMCW radar as the primary sensor for long-term mobile robot localization and navigation tasks. However, our current radar SLAM system depends on an expensive and cumbersome radar sensor. We aim to extend its application on some low-cost, lightweight radar sensors in future. For future work, we also seek to use the map built by our SLAM system and perform long-term localization on it across all weather conditions.

Acknowledgements
We thank Joshua Roe, Ted Ding, Saptarshi Mukherjee, Dr Marcel Sheeny and Dr Yun Wu for the help of our data collection.

Declaration of Conflicting Interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This

---

**Figure 26.** (a)–(c): Estimated trajectories and ground truth on multi-session traversals in normal, rain and snow conditions. (d) Top: image data in normal weather captured by our stereo camera, centre: image data in rain captured by our stereo camera, bottom: image data in snow captured by our phone for reference.

**Table 6.** Completion Percentage %.

| Dataset  | Oxford | MullRan | RADIATE |
|----------|--------|---------|---------|
| SuMa     | 20     | 72      | 63      |
| Baseline SLAM | 100 | 100     | 100     |
| ORB-SLAM2 | 90     | N/A     | N/A     |
| Our SLAM | 100    | 100     | 100     |

Completion Percentage on Different Datasets.

**Table 7.** Parameters for radar SLAM.

| Parameter | Value | Note |
|-----------|-------|------|
| Max polar distance | 87.5 | Maximum selected distance in radar reading in metres in our experiments |
| Min Hessian | 700 | Minimum Hessian value a point to be considered as keypoint |
| $\delta_c$ | 3 | Pixel value for maximal clique in graph outlier rejection in equation (5) |
| Max tracked points | 60 | Maximum number of points in tracking |
| Keyframe distance | 2.0 | Distance between keyframes in metres |
| Keyframe rotation | 0.2 | Rotation between keyframes in radians |
| $r_{pca}$ | 3 | PCA ratio to reject loop candidate in |

**Figure 27.** Computation time of tracking on the Rain sequence.
work was supported by EPSRC Robotics and Artificial Intelligence ORCA Hub (grant No. EP/R026173/1) and EU H2020 Programme under EU Marine Robots project (grant ID 731103).

ORCID iD
Sen Wang @ https://orcid.org/0000-0003-1537-8834

Supplementary Material
Supplementary material for this article is available online.

References

Aldera R, De Martini D, Gadd M, et al. (2019a). Fast radar motion estimation with a learnt focus of attention using weak supervision. In: International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019, IEEE, pp. 1190–1196.

Aldera R, De Martini D, Gadd M, et al. (2019b). What could go wrong? introspective radar odometry in challenging environments. In: IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27 October 2019, IEEE, pp. 2835–2842.

Aldibaja M, Suganuma N and Yoneda K (2016). Improving localization accuracy for autonomous driving in snow-rain environments. In: IEEE/SECE International Symposium on System Integration (SII), 13 October 2016, IEEE, pp. 212–217.

Almalioglu Y, Turan M, Lu CX, et al. (2021). Milli-rio: ego-motion estimation with low-cost millimetre-wave radar. IEEE Sensors Journal 21(3): 3314–3323.

Bailo O, Rameau F, Joo K, et al. (2018) Efficient adaptive non-maximal suppression algorithms for homogeneous spatial keypoint distribution. Pattern Recognition Letters 106: 53–60.

Barnes D, Gadd M, Murcutt P, et al. (2020a) The Oxford radar robotcar dataset: a radar extension to the oxford robotcar dataset. In: IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019, Paris: IEEE, pp. 6433–6438.

Barnes D and Posner I (2020) Under the radar: Learning to predict robust keypoints for odometry estimation and metric localisation in radar. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019, Paris: IEEE, pp. 9484–9490.

Barnes D, Weston R and Posner I (2020b) Masking by moving: Learning distraction-free radar odometry from pose information. In Conference on Robot Learning, Auckland, New Zealand, 14–18 December 2019, pp. 303–316.

Behley J and Stachniss C (2018) Efficient surfel-based slam using 3d laser range data in urban environments. In: Robotics: Science and Systems (RSS), New York, 1–27 July 2018.

Burnett K, Schoellig AP and Barfoot TD (2021) Do we need to compensate for motion distortion and doppler effects in spinning radar navigation? IEEE Robotics and Automation Letters 6(2): 771–778.

Campos C, Elvira R, Rodriguez JIG, et al. (2020) ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map slam. arXiv Preprint arXiv:2007.11898.

Carballo A, Lambert J, Monroy A, et al. (2020) LIBRE: The multiple 3d lidar dataset. arXiv Preprint arXiv:2003.06129.

Cen SH and Newman P (2018) Radar-only ego-motion estimation with millimeter-wave radar under diverse and challenging conditions. In: IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019, IEEE, pp. 1–8.

Cen S. H. and Newman P (2019) Radar-only ego-motion estimation in difficult settings via graph matching. In: IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019. IEEE, pp. 298–304.

Challis JH (1995) A procedure for determining rigid body transformation parameters. Journal of Biomechanics 28(6): 733–737.

Chandran M and Newman P (2006) Motion estimation from map quality with millimeter wave radar. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 9–15 October 2006. IEEE, pp. 808–813.

Charron N, Phillips S and Waslander SL (2018) De-noising of lidar point clouds corrupted by snowfall. In: Conference on Computer and Robot Vision (CRV), Toronto, ON, 8–10 May 2018, IEEE, pp. 254–261.

Chechpin P, Gérossier F, Blanc C, et al. (2010) Radar scan matching slam using the fourier-mellin transform. Springer Tracts in Advanced Robotics 62: 151–161.

Clark S and Durrant-Whyte H (1998) Autonomous land vehicle navigation using millimeter wave radar. In: IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019. IEEE, volume 4, pp. 3697–3702.

De Martini D, Gadd M and Newman P (2020) kradar++: Coarse-to-fine fmcw scanning radar localisation. Sensors 20(21): 6002.

Dissanayake WMG, Newman P, Clark S, et al. (2001) A solution to the simultaneous localization and map building (slam) problem. IEEE Transactions on Robotics and Automation 17(3): 229–241.

Engel J, Schöps T and Cremers D (2014)Lsd-slam: Large-scale direct monocular slam. In: European Conference on Computer Vision (ECCV), Zurich, Switzerland. Springer, pp. 834–849.

Gadd M, De Martini D and Newman P (2020) Look around you: Sequence-based radar place recognition with learned rotational invariance. In: IEEE/ION Position, Location and Navigation Symposium (PLANS), Portland, Oregon, 20–23 April 2020. IEEE, pp. 270–276.

Gadd M, De Martini D and Newman P (2021) Unsupervised place recognition with deep embedding learning over radar videos. arXiv Preprint arXiv:2106.06703.

Garg K and Nayar SK. (2004). Detection and removal of rain from videos. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 27 June–2 July 2004 . IEEE.

He L, Wang X and Zhang H (2016) M2dp: A novel 3d point cloud descriptor and its application in loop closure detection. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 9–15 October 2006. IEEE, pp. 231–237.

Himstedt M, Frost J, Hellbach S, et al. (2014). Large scale place recognition in 2d lidar scans using geometrical landmark relations. In: IEEE/RSJ International Conference on
Intelligent Robots and Systems, Chicago, IL, 14–18 September 2014. IEEE, pp. 5030–5035.

Holder M, Hellwig S and Winner H (2019) Real-time pose graph slam based on radar. In: IEEE Intelligent Vehicles Symposium (IV), Paris, France. IEEE, pp. 1145–1151.

Hong Z, Petillot Y. and Wang S. (2020) Radarslam: Radar based large-scale slam in all weathers. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 9–15 October 2006, pp. 5164–5170.

Howard A (2008) Real-time stereo visual odometry for autonomous ground vehicles. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 9–15 October 2006. IEEE, pp. 3946–3952.

Huang H, Sun Y and Liu M (2019) Reliable monocular ego-motion estimation system in rainy urban environments. In: IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand: 27-30 October 2019, pp. 1290–1297.

Jokela M, Kutila M and Pyykönä P (2019) Testing and validation of automotive point-cloud sensors in adverse weather conditions. Applied Sciences 9(11): 2341.

Jose E. and Adams M. D (2005) An augmented state slam formulation for multiple line-of-sight features with millimetre wave radar. In: IEEE/RSJ International conference on Intelligent Robots and Systems (IROS), Edmonton, AB, Canada: 2-6 August 2005, pp. 3087–3092.

Kim G, Park YS, Cho Y, et al. (2020) Mulran: multimodal range dataset for urban place recognition. In: IEEE International Conference on Robotics and Automation (ICRA), Paris, France: 31 May-31 August 2020, pp. 6246–6253.

Kone J and Janezic D (2007) An improved branch and bound algorithm for the maximum clique problem. Proteins 4(5): 590–596.

Kümmerle R, Grisetti G, Strasdat H, et al. (2011) g2o: A general framework for graph optimization. In: IEEE international conference on robotics and automation, Shanghai, China: 9-13 May 2011, pp. 3607–3613.

Li Y, Tan RT, Guo X, et al. (2016). Rain streak removal using layer priors. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA: 27-30 June 2016, pp. 2736–2744.

Lucas BD and Kanade T. (1981). An iterative image registration technique with an application to stereo vision. In: International Joint Conference on Artificial Intelligence (IJCAI), Vancouver, B.C., Canada: 24-28 August 1981.

Maddern W, Pascoe G, Linegar C, et al. (2017) 1 Year, 1000km: the Oxford robotCar dataset. The International Journal of Robotics Research 36(1): 3–15.

Mark JW, Mohamoud A, vd Houwen E, et al. (2013) Indoor radar slam a radar application for vision and gps denied environments. In: European radar conference, Nuremberg, Germany: 9-11 October 2013, pp. 471–474.

Mur-Artal R, Montiel JMM and Tardos JD (2015) Orb-slam: a versatile and accurate monocular SLAM system. IEEE Transactions on Robotics 31(5): 1147–1163.

Mur-Artal R and Tardós JD (2017) ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. IEEE Transactions on Robotics 33(5): 1255–1262.

Ort T, Gilitschenski I and Rus D (2020) Autonomous navigation in inclement weather based on a localizing ground penetrating radar. IEEE Robotics and Automation Letters 5(2): 3267–3274.

Park YS, Kim J and Kim A (2019) Radar localization and mapping for indoor disaster environments via multi-modal registration to prior lidar map. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 9–15 October 2006. IEEE, pp. 1307–1314.

Park YS, Shin Y-S and Kim A (2020) Pharao: Direct radar odometry using phase correlation. In: IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019. IEEE, pp. 2617–2623.

Porav H, Bruls T and Newman P (2019) I can see clearly now: Image restoration via de-raining. In: IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019. IEEE, pp. 7087–7093.

Qin T, Li P and Shen S (2018) Vins-mono: a robust and versatile monocular visual-inertial state estimator. IEEE Transactions on Robotics 34(4): 1004–1020.

Ren W, Tian J, Han Z, et al. (2017). Video desnowing and de-raining based on matrix decomposition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA: 21-26 July 2017, pp. 4210–4219.

Rouveure R, Monod M and Faure P (2009) High resolution mapping of the environment with a ground-based radar imager. In: International Radar Conference Surveillance for a Safer World (RADAR), Bordeaux, France: 12-16 October 2009. IEEE, pp. 1–6.

Săftescu Ş, Gadd M, De Martini D., et al. (2020) Kidnapped radar: Topological radar localisation using rotationally-invariant metric learning. In: IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019. IEEE, pp. 4358–4364.

Schubert D, Demmel N, Usenko V, et al. (2018). Direct sparse odometry with rolling shutter. In: European Conference on Computer Vision (ECCV).

Schuster F, Keller CG, Rapp M, et al. (2016). Landmark based radar slam using graph optimization. In: IEEE International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil: 1-4 Nov. 2016. IEEE, pp. 2559–2564.

Shan T and Engelb B (2018) Lego-loam: lightweight and ground-optimized lidar odometry and mapping on variable terrain. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 9–15 October 2006. IEEE, pp. 4758–4765.

Shan T, Engelb B, Meyers D, et al. (2020) Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 9–15 October 2006. IEEE, pp. 5135–5142.

Sheeny M, Pellegrin ED, Mukherjee S, et al. (2021). Radiate: a radar dataset for automotive perception. In: IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019. IEEE.

Solá J, Deray J and Atchuthan D (2018) A micro lie theory for state estimation in robots. CoRR, abs/1812.01537.
Tang TY, De Martini D, Barnes D, et al. (2020a) Rsl-net: localising in satellite images from a radar on the ground. *IEEE Robotics and Automation Letters* 5(2): 1087–1094.

Tang TY, De Martini D, Wu S, et al. (2020b) Self-supervised localisation between range sensors and overhead imagery. *Robotics: Science and Systems (RSS) 12th-17th July 2020.*

Vivet D, Checchin P and Chapuis R (2012) Radar-only localization and mapping for ground vehicle at high speed and for riverside boat. In: IEEE International Conference on Robotics and Automation (ICRA), Montreal, Canada, 20–24 May 2019. IEEE, pp. 2618–2624.

Vivet D, Gérossier F, Checchin P, et al (2013) Mobile ground-based radar sensor for localization and mapping: an evaluation of two approaches. *International Journal of Advanced Robotic Systems* 10(8): 307.

Wang W, de Gusmo PP, Yang B, et al (2021) Learning to relocalize in fmcw radar. *arXiv Preprint arXiv:2103.11562.*

Yamada M, Sato T, Chishiro H, et al. (2019) Vision-based localization using a monocular camera in the rain. In: IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand: 27-30 October 2019. IEEE, pp. 293–298.

Yoneda K., Hashimoto N., Yanase R., et al (2018) Vehicle localization using 76ghz omnidirectional millimeter-wave radar for winter automated driving. In: IEEE Intelligent Vehicles Symposium (IV), Changshu, China: 26-30 June 2018. IEEE, pp. 971–977.

Zhang C, Ang MH and Rus D (2018) Robust lidar localization for autonomous driving in rain. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 9–15 October 2006. IEEE, pp. 3409–3415.

Zhang J and Singh S. (2014). Loam: Lidar odometry and mapping in real-time. In: Robotics: Science and Systems (RSS), Rome, Italy: 13-15 July, 2015.

Zhang Z and Scaramuzza D (2018) A tutorial on quantitative trajectory evaluation for visual(-inertial) odometry. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)), Beijing, China, 9–15 October 2006. IEEE, pp. 7244–7251.