

---

# GROUNDDED: A Localizing Ground Penetrating Radar Evaluation Dataset for Learning to Localize in Inclement Weather

Journal Title  
XX(X):1–17  
©The Author(s) 2023  
Reprints and permission:  
sagepub.co.uk/journalsPermissions.nav  
DOI: 10.1177/ToBeAssigned  
www.sagepub.com/  
**SAGE**

Teddy Ort<sup>1</sup>, Igor Gilitschenski<sup>2</sup>, and Daniela Rus<sup>1</sup>

## Abstract

Mapping and localization using surface features is prone to failure due to environment changes such as inclement weather. Recently, Localizing Ground Penetrating Radar (LGPR) has been proposed as an alternative means of localizing using underground features that are stable over time and less affected by surface conditions. However, due to the lack of commercially available LGPR sensors, the wider research community has been largely unable to replicate this work or build new and innovative solutions. We present GROUNDED, an open dataset of LGPR scans collected in a variety of environments and weather conditions. By labeling these data with ground truth localization from an RTK-GPS / Inertial Navigation System, and carefully calibrating and time-synchronizing the radar scans with ground truth positions, camera imagery, and Lidar data, we enable researchers to build novel localization solutions that are resilient to changing surface conditions. We include 108 individual runs totalling 450 km of driving with LGPR, GPS, Odometry, Camera, and Lidar measurements. We also present two new evaluation benchmarks for 1) Localizing in Weather and 2) Multi-lane Localization, to enable comparisons of future work supported by the dataset. Additionally, we present a first application of the new dataset in the form of LGPRNet: an inception-based CNN architecture for learning localization that is resilient to changing weather conditions. The dataset can be accessed at <http://lgprdata.com>.

## Keywords

LGPR, Ground Penetrating Radar, Autonomous Vehicles, Deep Learning for Robotics, Localization, Mapping, Driving in Weather

## Introduction

The ability to localize in the environment is critical to enable the widespread deployment of autonomous vehicles. While the Global Positioning System (GPS) is often used to obtain approximate global localization, it lacks the accuracy necessary to meet the stringent requirements of autonomous driving Wing et al. (2005). For this reason, most fielded autonomous vehicle solutions currently localize on HD maps with either lidar sensors Levinson et al. (2007); Levinson and Thrun (2010); Wolcott and Eustice (2015), cameras Mur-Artal et al. (2015), or both Wolcott and Eustice (2014). Localization with these sensors can provide accuracy in the range needed for autonomous operation. However, sensors that rely heavily on surface features in the environment have an inherent failure mode should the environment change between the mapping and localization phases. Some approaches aim to filter out dynamic objects during mapping Bescos et al. (2018). Others seek to

identify and map only stable features or landmarks in the environment Dymczyk et al. (2016); Bürgi et al. (2019). Robustly dealing with inclement weather such as snow is particularly challenging, as snowfall can dramatically alter the surface appearance. Solving this problem remains one of the open challenges to enable human-level (or above) performance of autonomous vehicles in diverse environments.

Recently, Localizing Ground Penetrating Radar (LGPR) Cornick et al. (2016) has been proposed to address the localization task in such environments. By mapping and localizing using features beneath the ground,

---

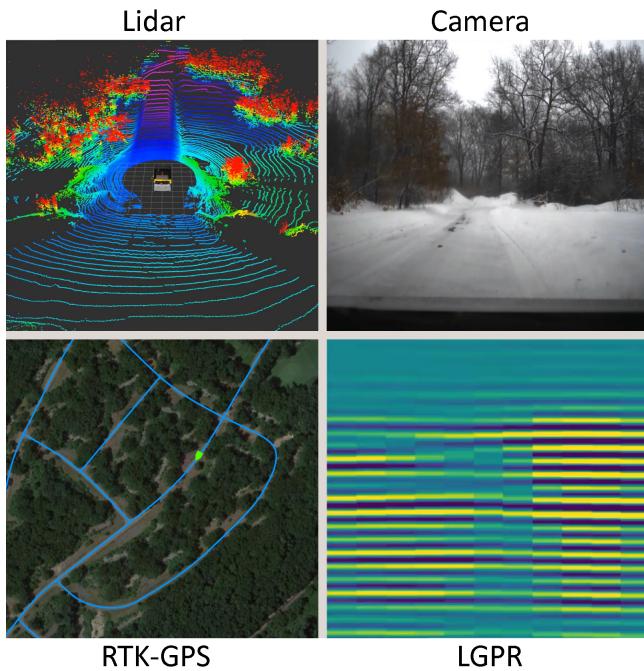
<sup>1</sup>Massachusetts Institute of Technology, US

<sup>2</sup>University of Toronto, CA

### Corresponding author:

Teddy Ort, Massachusetts Institute of Technology, US

Email: teddy@mit.edu



**Figure 1.** The GROUNDED dataset includes four data streams for each run. 1) Lidar scans from a roof-mounted Velodyne HDL-64, 2) Camera images from a front-facing Point Grey Grasshopper camera, 3) Base station-corrected RTK-GPS for groundtruth, and 4) LGPR data stream from the rear-mounted radar unit.

LGPR can avoid the instability of surface-based maps. LGPR was used as the sole localization sensor in Ort et al. (2020) to navigate an automated vehicle in clear weather, rain, and snow conditions. However, the sensor demonstrated in that work is not commercially available. Typical Ground Penetrating Radar (GPR) sensors utilize frequencies in the 0.5 GHz to 2.5 GHz range Benedetto et al. (2017) to obtain a high-resolution image of objects close to the surface. However, a lower frequency of 100 MHz to 400 MHz is ideal for localization because the greater penetration depth enables mapping of deeper, more stable features Cornick et al. (2016). Several companies have recently announced plans to commercialize LGPR technology Ryan (2017); Shaw (2020). Currently, however, there are still no options to purchase LGPR sensors suitable for localization. The lack of publicly available sensors or datasets has left the technical community in a holding pattern.

In this work, we aim to address the lack of access to LGPR systems and to enable algorithmic development for localization of autonomous vehicles in a wide range of weather conditions and illumination. We describe and release an open dataset of LGPR frames collected using one of the prototype sensors described in Cornick et al. (2016). We also propose two challenges aimed at accelerating the development of solutions for mapping and localization under challenging driving conditions such as difficult illumination,

heavy rain, and snow. We believe the dataset will enable the research community to replicate and improve upon the current state-of-the-art, and to tackle new open problems for autonomous driving in difficult weather. For example, Ort et al. (2020) found a degradation in localization performance when localizing in rain or snow, perhaps due to the unmodeled changes to the moisture content of the soil. Furthermore, both Cornick et al. (2016) and Ort et al. (2020) conducted both mapping and localization in a single lane. Since practical autonomous vehicles will need to change lanes, it is an important area of research to stitch together multiple lanes to form a coherent map.

The dataset consists of 108 runs amounting to a total of 450 km and 12 hours of driving. For each route, there is data associated with clear, rainy, and snowy weather (see Fig. 2). The data includes groundtruth GPS location, odometry, and scans from the LGPR sensor, the camera, and the lidar system on the vehicle. Because the LGPR sensor has a data collection footprint equal to its width, which is smaller than the width of a road, a single path of the LGPR does not provide a complete map of the ground features for that road. We address this limitation by providing multiple paths for each segment, with the vehicle driving left, center, and right on the road (see Fig. 3), along with a challenge to align and stitch different paths into a complete map.

In summary, this paper contributes the following:

- The first publicly available dataset of Ground Penetrating Radar data for localization and mapping collected in a variety of weather conditions, and multiple adjacent lanes;
- Two challenge benchmarks: 1) Localization in Weather and 2) Multi-lane Localization to compare LGPR research;
- Additional lidar and camera data streams to enable comparison with existing visual and lidar navigation approaches in driving.

The remainder of this paper is organized as follows: In the next section we review related work with LGPR sensors and autonomous driving datasets. In Sec. **Benchmark Challenges** we present the challenge benchmarks. Next, in Sec. **Dataset** we describe the dataset organization and software development kit. Then, in Sec. **Data Collection Platform** we describe the research platform, sensor suite, calibration, and time synchronization of the sensor streams. Finally, in Sec. **Conclusion** we conclude with some final thoughts regarding promising directions for future research.

## Related Work

**Model-based Localization:** Over the last two decades, the robotics community extensively considered the problem of localization and mapping Cadena et al. (2016) involving a diverse set of sensors, most prominently cameras Lowry et al. (2016) and lidars Levinson et al. (2007); Levinson and Thrun (2010); Wolcott and Eustice (2015). Particularly for autonomous driving, a considerable amount of work focused on dealing with challenging and changing appearance conditions Bürki et al. (2019) such as weather Doan et al. (2019) or occlusions Bescos et al. (2018); Fehr et al. (2017). To improve robustness, radar has also been considered as a localization modality for autonomous driving Burnett et al. (2021); Rapp et al. (2015); Werber et al. (2016, 2019). Even with this additional modality, robust localization remains challenging due to phenomena such as occlusions. The goal of this dataset is to overcome or completely avoid some of these challenges. It enables wider research on a complementary localization modality which does not suffer from occlusion by dynamic objects and changes in appearance conditions.

**Learning-based Localization** Learning-based localization methods such as Kendall et al. (2015); Liu et al. (2020) aim to learn a model for localizing a robot on a prior dataset of surface images. Similarly, our proposed learning-based approach, LGPRNet extends these to underground features. However, unlike surface features, which can be viewed from many different poses, underground radar features are visible only from the region immediately above them, which substantially affects the localization problem.

**Ground Penetrating Radars:** Only a few works have considered the use of ground penetrating radars in robotics, such as for landmine detection Dawson-Howe and Williams (1998) or for autonomous surveys Williams et al. (2012). Using GPRs for localization has so far been considered only in Cornick et al. (2016); Ort et al. (2020). Consequently, most GPR datasets are targeted at very different application domains, e.g., for research on soil structure characterization Romero-Ruiz et al. (2018) or meteorology Kubota et al. (2009). Recently, Baikovitz et al. (2021) learned relative GPR models to correct for odometry drift using a factor graph. However, since the GPR sensor was single-channel, it wasn't suitable for localization to a prior map. There is currently no dataset allowing for widespread localization research with GPRs. The high cost of GPRs and mere prototype availability of GPRs specifically designed for localization makes research in this field completely impossible for many groups. By making

their data publicly available, the authors aim to overcome this limitation, simplifying research on radiogeological navigation.

**Datasets:** Because of the high cost of a retrofitted autonomous vehicle and to compare results more equally, a lot of Autonomous Driving research is already driven by benchmark datasets. In that context, the KITTI Geiger et al. (2013) dataset is one of the earliest and most popular in autonomous driving research. In recent years, numerous institutions made the data from their research vehicles publicly available Cordts et al. (2016); Huang et al. (2018); Maddern et al. (2017); Pitropov et al. (2020); Sun et al. (2020) some of which also involve radar data Barnes et al. (2020); Caesar et al. (2020). A dataset specifically focusing on radar perception is presented in Meyer and Kuschk (2019). More recently, several datasets covering novel sensing modalities have been made publicly available, focusing on acoustic detection Schulz et al. (2021) and dynamic vision sensors (DVS) Binns et al. (2017); Zhu et al. (2018). In the same spirit, our work contributes ground penetrating radars as a new sensing modality to dataset-driven perception research.

## Benchmark Challenges

Prior localization results with LGPR have looked promising, yet there are two important limitations that must be overcome before LGPR sensors can be practically useful. The first requirement is to devise algorithms that can localize even when the prior map was recorded in different weather conditions. This can be challenging because LGPR data can be affected by the moisture content and temperature of the underground soil, which can vary with surface weather conditions. In Ort et al. (2020) a degradation in localization performance in rain and snow was measured, but their algorithm did not explicitly account for weather changes. The second requirement is to build maps that can localize a vehicle while it is changing between multiple lanes. Since the LGPR sensor only records data directly beneath it, and the sensor only spans the width of the vehicle, prior work Cornick et al. (2016); Ort et al. (2020) only used maps consisting of a single lane. Since practical autonomous vehicles will need the ability to maintain a seamless localization as they traverse multiple lanes, it will be necessary to devise mapping and localization algorithms that can stitch together data from multiple passes in different lanes to obtain a cohesive road map. To ensure solutions to these limitations can be compared on an equal footing, we propose the following two challenge benchmarks. Our

Year	Dataset	Location	Weather	Camera	Lidar	Radar	Other
2013	KITTI Geiger et al. (2013)	Karlsruhe	dry	✓	✓	-	
2016	Cityscapes Cordts et al. (2016)	50 cities	dry	✓	-	-	
2017	DDD17 Binas et al. (2017)	Switzerland & Germany	dry, rain	✓	-	-	DVS
2017	Oxford Maddern et al. (2017); Barnes et al. (2020)	Oxford	dry, rain, snow	✓	✓	✓	
2018	ApolloScape Huang et al. (2018)	4 x China	dry, rain, snow	✓	-	-	
2018	MVSEC Zhu et al. (2018)	Philadelphia	-	✓	✓	-	DVS
2019	Astyx Meyer and Kuschk (2019)	-	-	✓	✓	✓	
2020	CADC Pitropov et al. (2020)	Waterloo	dry, snow	✓	✓	-	
2020	nuScenes Caesar et al. (2020)	Boston, Singapore	dry, rain	✓	✓	✓	
2020	Waymo Sun et al. (2020)	San Francisco, Phoenix, Detroit, LA, Seattle, Mountain View	dry, rain, snow	✓	✓	-	
2021	Delft Schulz et al. (2021)	Delft	-	✓	-	-	Mic. array
2021	<b>Ours</b>	Massachusetts	dry, snow, rain	✓	✓	LGPR	

**Table 1.** Overview of exteroceptive sensing modalities in autonomous vehicle navigation research datasets.

dataset specifically includes data to address these challenges, including data collected in a variety of weather conditions and in multiple lanes, as shown in Table 3.

### Localization in Weather Challenge

**Mapping and Localization Runs** In the provided dataset, every run was collected as a pair to enable mapping and localization using the same environmental conditions. For example, *run\_0001* and *run\_0002* were both collected driving the same route, in an urban environment, in clear weather and in the right lane. All of this information can be found in the *runs.csv* file, as described in Sec. Run Level Data. However, for the Localization in Weather Challenge, we aim to evaluate localization using maps that were created in different weather to demonstrate weather resilience. Therefore, we could instead evaluate localization using *run\_0037* or *run\_0038*, which were both collected along the same route, but in snowy weather, while still using *run\_0001* or *run\_0002* to build the map. In short, each run in the dataset includes a weather condition label [*clear*, *rain*, *snow*]. For this challenge, the mapping and localization runs should be along the same geographical route, but in different weather conditions.

In the event learning-based algorithms are utilized for mapping, we would like to clarify here that it is acceptable to include runs which were collected in inclement weather in the training phase, as long as they were collected at a different location. This mirrors the real-world constraints where one could envision training an algorithm to localize

to single weather maps in all weather by including a small sample of varying weather in the training phase. Finally, while other onboard sensors (e.g. odometry, camera, or lidar) may be used to enhance the localization estimate, the estimate at each time  $t_i$  may only use measurements taken at a prior time  $t_j$  such that  $j \leq i$ .



**Figure 2.** Left: Camera images from three different runs of the same route (route\_04) capturing LGPR data in the same location in clear weather, snow, and rain to support the Localization in Weather Challenge. Right: Trajectory of route\_04 overlaid on a map. Other runs capture a variety of environments including rural (shown here), urban, and highway.

**Localization in Weather Metric** The metric used for localization in weather is an Absolute Trajectory Evaluation. Since the LGPR data is labeled with ground truth RTK-GPS data, we can directly evaluate the trajectory error rather than implementing a relative metric, as is often needed with SLAM or VIO solutions Zhang and Scaramuzza (2018). Additionally, we limit the benchmark to the two translation degrees of freedom and single rotation (yaw) on the surface of the ground, as these are the critical values necessary for autonomous driving. Thus, the first two metrics are the Root Mean Square Error (RMSE) of the translation and orientation of the vehicle  $t_{rmse}, \theta_{rmse}$  evaluated over the entire run. Next, because for driving purposes, the lateral translation error (with respect to the lane) is often far more important than the longitudinal, we further decompose the error into its lateral and longitudinal components to obtain two additional error metrics:  $t_{lat}, t_{long}$ . While all four of these error metrics should be reported, for the purpose of providing an overall score we propose a weighting:

$$s = t_{lat} + 0.1t_{long} + 10\theta_{rmse} \quad (1)$$

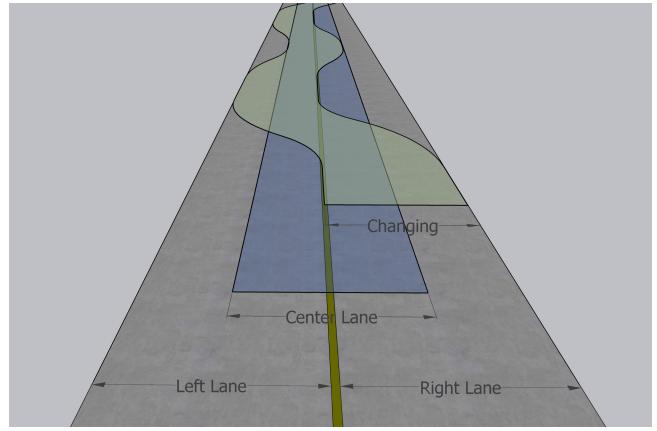
Intuitively, this assigns equal cost to 10 cm of error in the lateral lane direction, 1 m of error in the longitudinal lane direction, and 0.57° of orientation error which have similar real-world importance. As a first step toward addressing this challenge, see Sec. [LGPRNet: Learning to Localize in Weather](#), where we propose a learning-based method for mitigating the effect of changing weather conditions on localization accuracy and evaluate it using this benchmark.

### Multi-lane Localization Challenge

**Mapping and Localization Runs** In the provided dataset, each run is labeled with a lane attribute in [Left, Center, Right, Changing] as shown in Fig. 3. These lanes are overlapping since the center lane is not a separate lane, but rather the sensor centerline is passing over the lane divider and including partial data from each of the left and right lanes. The purpose of these runs is specifically to support building cohesive maps that can track a vehicle even while it is changing lanes, or only partially overlapping a lane. For this challenge, multiple runs where the lane is in [Left, Center, Right] in the same route should be used for map creation. In runs where the lane is marked *Changing*, the vehicle was driven along the route while randomly choosing any of the [Left, Center, Right] lane positions and changing between them. Those runs should only be used for localization evaluation and not included in mapping or training data.

**Multi-lane Localization Metric** The metric used for the Multi-lane Localization benchmark is similar to that described for the previous benchmark. One important difference is that here we explicitly do not follow a single lane. Therefore, lateral and longitudinal lane errors are not needed and only the absolute trajectory errors  $t_{rmse}, \theta_{rmse}$  are necessary. The overall score is then calculated as:

$$s = t_{rmse} + 10\theta_{rmse} \quad (2)$$



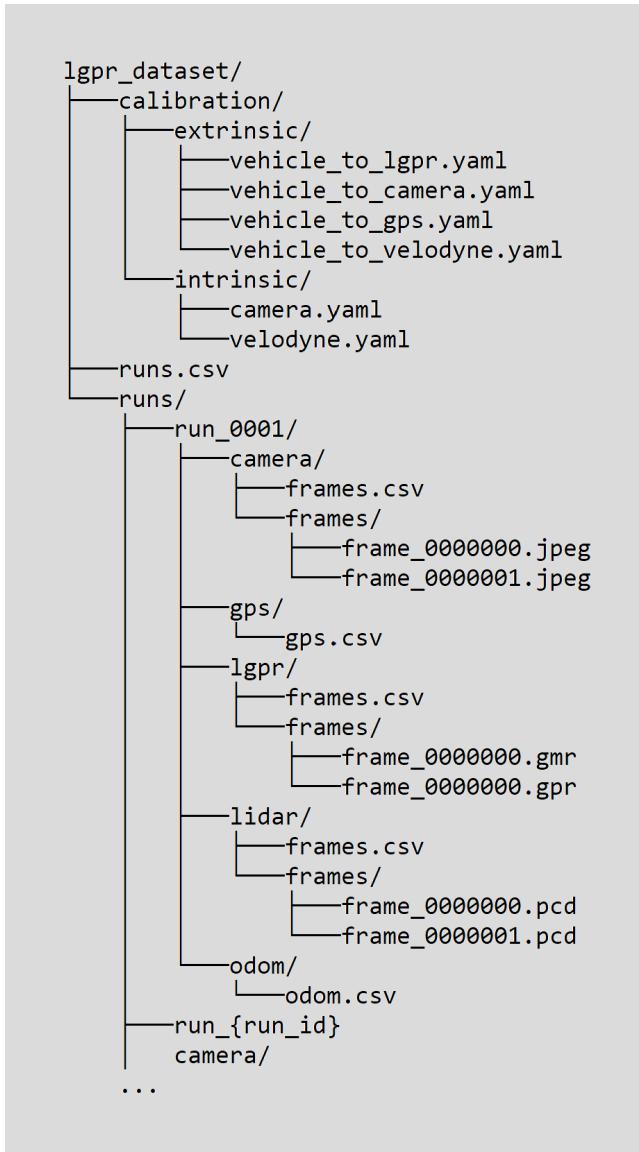
**Figure 3.** The Multi-lane Localization Challenge provides LGPR frames collected in four lane positions: 1) Left, 2) Right, and 3) Center 4) Changing. These runs can be used to create a consistent map capable of localizing the vehicle continuously even while switching lanes.

### Dataset

The dataset is available for download at <http://lgprdata.com>. The data is stored hierarchically, as shown in Fig. 4.

### Run Level Data

At the highest level are runs. Each run is a single data collection instance often taking the form of a loop. To avoid potential issues with overlapping sensor data, the start and end points of the loop always have a small gap of  $\approx 10$  m. For each run, we provide run-level information in a file called *runs.csv*. This file contains columns as shown in Table 2. Each row in *runs.csv* has a unique *run\_id* which corresponds to a directory such as *run\_0001* in the *runs* directory (see Fig. 4). Note that the *route\_id* corresponds to a unique route traversal but duplicates do exist because the same route was driven in multiple runs. For example, *run\_0001* and *run\_0002* represent two unique runs, but have the same *route\_id* because they traversed the same physical route. Furthermore, a single route can be traversed in any of the lanes, or in either direction. One final caveat to bear in mind



**Figure 4.** The dataset is organized in directories for each run. Run metadata such as the lane and weather conditions is provided in the *runs.csv* file and referenced by the corresponding *{run.id}*. Similarly, for each sensor, frame metadata is provided in a *frames.csv* file referencing individual frames by their *{frame.id}*.

is that the direction of traversal rotates the semantic meaning of the lane column. Thus, two runs with the same *route\_id* but different direction values would imply traversing the same lane in opposite directions if the lane value in one was equal to *right* while in the other was *left*. In general, the purpose of grouping runs by route, lane, and direction, is to provide data for the Multi-lane Localization Challenge (Sec. [Multi-lane Localization Challenge](#)). For the simple case of mapping and localizing on the same trajectory (as in the prior work), one can simply use runs with identical route, lane, and direction.

The dataset contains runs in a variety of lanes, and environmental and weather conditions to support the benchmark challenges proposed earlier. Table 3 shows the splits in the data for each of the relevant data conditions.

Column	Data Type	Description
run_id	Integer	A unique value for each run in the dataset
date	Date	The date when the run was recorded
road_type	String	One of {urban, rural, highway}
route_id	Integer	A unique value for each route
weather	String	One of {clear, rain, snow}
direction	Integer	$\{-1, 1\} = \{\text{clockwise}, \text{anticlockwise}\}$
lane	String	One of {left, center, right, changing}
length	Float	The total length (km) of the run
duration	Float	The total duration (s) of the run
sensors	List	A list of comma separated sensor names

**Table 2.** The file *runs.csv* contains metadata for every run in the dataset organized into the columns shown.

### Sensor Level Data

Within each run directory, there are several directories, one for each element in the sensors field in the corresponding row of *runs.csv*. Currently, every run includes at a minimum [*lgpr*, *gps*, *odom*]. Many runs additionally include [*camera*, *lidar*]. Next, we briefly describe the data formats of each of these sensor streams; for additional details, please see the dataset documentation.

**lgpr** The LGPR sensor frames are arrays with shape (11, 369) corresponding to the 11 radar channels (11 Tx and Rx pairs for the 12 radar elements in the array) and the 369 depth bins. Each value in the array is an *int8* ranging from [-128, 127]. The *Lgpr* directory contains a file *frames.csv* which includes a table containing the *frame\_id* for each LGPR frame in the run, along with a timestamp providing the synchronized time (see Sec. [Time Synchronization](#)) for that frame. In the *frames* directory, two files are included for each frame in *frames.csv*: 1) *frame\_id.gpr* contains the raw LGPR array in csv format while 2) *frame\_id.gmr* contains the processed LGPR scan with mean removed, as described in Sec. [Calibration](#). A script for reading and visualizing these scans is provided in the SDK.

**gps** The ground truth GPS data includes position and velocity. Each GPS reading contains sixteen values including [*longitude*, *latitude*, *altitude*], position and orientation in the UTM frame [*x*, *y*, *z*, *qx*, *qy*, *qz*, *qw*], and velocity in the vehicle frame [ $\dot{x}$ ,  $\dot{y}$ ,  $\dot{z}$ ,  $\dot{r}$ ,  $\dot{p}$ ,  $\dot{q}$ ]. Note that most GPS measurements include Differential GPS (DGPS) corrections received from a fixed base station. However, because our base station has a range limited to  $\approx 10$  km, runs with *road\_type* equal

to *highway* in the *runs.csv* file do not have base station corrections. All of these twelve columns are included, along with a timestamp in a file called *gps.csv* which includes all GPS measurements for the run.

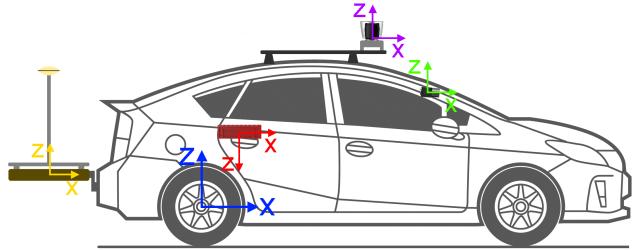
***odom*** The odometry data includes measurements from the wheel encoders on each of the rear wheels and the Inertial Measurement Unit (IMU). For each wheel encoder measurement we include the total distance in meters, traversed by the left and right rear wheels respectively along with a timestamp. These measurements are provided in *encoder.csv*. Next, the IMU data is a vector of ten values including 3-vectors for linear acceleration and angular velocity, and an orientation quaternion. These values, along with a timestamp, are provided for each IMU measurement. Finally, for convenience, we also provide *odom.csv*, which contains the vehicle position  $[x, y, z]$  and orientation quaternion  $[qx, qy, qz, qw]$  with respect to the start frame calculated by accumulating the wheel encoder and IMU measurements using a Kalman filter.

***camera*** The LGPR sensor is particularly useful in situations where camera sensors can fail such as glare, darkness, or inclement weather. For this reason, we include camera data for comparison purposes. We use a PointGrey Grasshopper front-facing camera mounted behind the windshield. Each frame has resolution 1928x1448. The file *camera.csv* contains a table with columns for the *frame\_id* of each camera frame along with the timestamp. The directory *frames/* contains camera images of the form *frame\_id.png*, where each image corresponds to the *frame\_id* in the *frames.csv* file.

***lidar*** Similar to the camera sensor, we provide lidar data primarily as a baseline comparison tool. The lidar sensor is a Velodyne HDL-64 mounted above the roof of the vehicle offset toward the front (see Fig. 5). It provides pointcloud data, where each scan contains  $\approx 191,000$  data points, and each point is a 4-vector of  $[x, y, z, intensity]$  giving the geometric position of the point with respect to the sensor frame, and the intensity of the laser reflection. We accumulate measurements until a  $360^\circ$  rotation is completed at a rate of 10 Hz and include one scan/revolution. The file *lidar.csv* contains a table with a column *frame\_id* along with a timestamp. The directory *frames/* contains files *frame\_id.pcd* which include the pointclouds in PCD format **Rusu and Cousins (2011)**.

	Type	Total Length (km)
<b>Road Type</b>	highway	316.2
	rural	115.2
	urban	17.6
<b>Weather</b>	clear	151.8
	rain	145.4
	snow	151.8
<b>Lane</b>	center	21.6
	changing	16.8
	left	38.4
	right	372.2

**Table 3.** The GROUNDED dataset contains data collected in a variety of lane positions, and environment and weather conditions to support the proposed benchmark challenges.



**Figure 5.** Top: The Toyota Prius vehicle platform used to collect the dataset. Bottom: A schematic showing the positions of reference frames for the vehicle and each of the sensors. Blue is the vehicle frame, Red is the OXTS GPS/IMU, Green is the camera, Magenta is the lidar, and Yellow is the LGPR sensor. The measured transforms between these frames are included in the dataset.

## Data Collection Platform

### Vehicle Infrastructure

The dataset was collected using a Toyota Prius research platform shown in Fig. 5. The sensors mounted on the vehicle include wheel encoders, an IMU, GPS, lidar, camera, and the LGPR radar array. The origin frame for the vehicle is located in the center of the rear axle, with the x-axis parallel to the ground and oriented toward the front of the vehicle, and the z-axis oriented upward. Fig. 5 shows the reference frames for all of the other sensors as well. Note that all of the sensors are located on the central plane of the vehicle (XZ-plane) except for the GPS unit, which is slightly offset. The actual values for all of these transforms are provided in the *calibration/extrinsic* directory, as shown in Fig. 4. In

the following subsections, we describe the details of each sensor and how the data was collected, synchronized, and calibrated.

The data was collected on two computers. The first one, *CAR\_PC*, was connected to a vehicle-wide LAN which received data from the GPS, IMU, wheel-encoders, and lidar. It also utilized a USB hub to receive images from the camera. Due to technical limitations of the LGPR sensor prototype, it could not be configured to stream the raw data to the vehicle LAN in real time. Instead, a second computer, *LGPR\_PC*, was used to receive and store the LGPR frames separately. In Sec. [Time Synchronization](#) we describe how we utilize an accurate GPS time reference to provide time-synchronized data streams. The *CAR\_PC* is a Dell Laptop with an i7 processor running Ubuntu 18.04. The *LGPR\_PC* is a Single Board Computer (SBC) onboard the LGPR sensor prototype and accessed via Ethernet for data retrieval.

## Sensors

The data collection vehicle platform included lidar, camera, radar, odometry, and gps senosors. Table 4 includes the models and key features of theses sensors. The following paragraphs also describe in detail how theses sensors were mounted and configured.

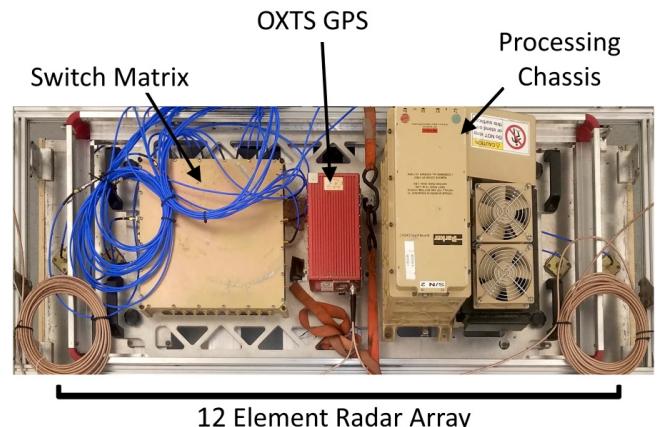
**LGPR Sensor** The LGPR sensor was mounted on the rear of the vehicle by attaching to the trailer hitch. It is a completely self-contained unit only connected to the vehicle for power. Here we describe the main components of the LGPR sensor; for more details see [Cornick et al. \(2016\)](#), which describes the design of the sensor. This sensor is not commercially available and there are only a few prototypes, which is the primary motivation for sharing the data in this work.

As seen in Fig. 6, the main LGPR sensor components include a 12-element radar array, a switch matrix, an OXTS RTK-GPS unit, and a processing chassis. The radar array transmits on a single element at a time, while receiving on the next element. Each pair of elements thus provides a single channel of data which produces images 11 pixels wide. Note that the array is 152 cm wide (the same as the width of the vehicle) and 61 cm from front to back. While the GPS unit contained in the LGPR sensor is typically used for onboard mapping, here we use it only for time-synchronization, instead using the more accurate base station-corrected GPS measurements taken onboard the vehicle for groundtruth, as described in Sec. [RTK-GPS](#)

At runtime, the array completes a sweep of all 11 channels at 126 Hz and these data are binned into 369 depth bins to provide an 11x369 pixel image that spans the width of the

Camera	Pointgrey Grasshopper3
Resolution	2736 x 2192
Sensor	Sony ICX694
Shutter	Global
Megapixels	6
Lidar	Velodyne HDL-64
Num. Channels	64
VFOV	26.9°
HFOV	360°
Data Rate	2.2M points/sec
Range	120m
GPS	OXTS RT3003
Correction Source	RTK Base Station
Positional Accuracy	0.01 m RTK
Roll / Pitch Accuracy	0.03°
Heading Accuracy	0.05°
Rate	100 Hz
Num Antennae	2
Radar	Custom LGPR Sensor
Mount	Tow-hitch
Num. Channels	11
Num. Radar Elements	12
Discrete Depths	369
Sensor Dimensions	152x61x7.6 cm
Penetration Depth	2-3m

**Table 4.** An overview of key features for the sensors included on the autonomous vehicle platform. Additional information can be found by consulting the specifications provided in the manufacturer datasheets.



**Figure 6.** The LGPR sensor used to collect the dataset. The processing chassis communicated with the switch matrix to control transmit and receive on the 12 radar elements. The GPS shown here was used for time synchronization, but groundtruth information was obtained by a separate GPS unit onboard the vehicle.

vehicle and penetrates approximately 3 m beneath the ground (the actual depth can vary by region and soil content). The switch matrix receives these signals and transmits them to the processing chassis, where the radar images are stored for later retrieval.

**RTK-GPS** An OXTS RT3003 Inertial Navigation System (INS) was used to provide a groundtruth vehicle position for all runs. To achieve an accuracy of  $\approx 2$  cm, the GPS sensor requires corrections to the raw GPS signal from a fixed base station. Note that for all runs with *road\_type* equal to *rural* or *urban* we placed and calibrated the base station in the test region as described in Sec. [Calibration](#). However, since the base station range is limited to 10 km, the runs with *road\_type* equal to *highway* do not include base station corrections.

**Odometry** The odometry sensors include two magnetic hall effect wheel encoders, one in each of the rear wheels. These provide counts as each pole passes the sensor. Additionally, we utilize the IMU built into the RT3003 to obtain acceleration and angular rates. Together, the wheel odometry and IMU data can be used to obtain an interoceptive estimate of the vehicle motion independent of the exteroceptive measurements of the GPS and LGPR sensors.

**Camera** We utilize a front-facing Point Grey Grasshopper camera with a resolution of 1928x1448 at 6fps. The camera is mounted behind the windshield to protect it from rain or snow. Windshield wipers were utilized when required to ensure the windshield remained clear. The main motivation for providing camera imagery in this work is to serve as a baseline to compare how weather affects vision sensors compared to LGPR.

**Lidar** A Velodyne HDL-64 sensor is mounted on the roof of the vehicle. We mount the sensor on the center plane of the vehicle, slightly forward to obtain a better view of the road in front of the vehicle at the expense of some occlusion caused by the vehicle itself in the rear. The Velodyne scanner spins at 600 RPM, yielding 360-degree scans at 10 Hz. The scans are labeled with the synchronized time at the end of each revolution; to obtain the time of individual points, one can linearly interpolate between the time stamps for each azimuth.

### Time Synchronization

As noted in Sec. [Vehicle Infrastructure](#), the LGPR sensor data is collected in isolation from the other vehicle sensors. We utilize the GPS time reference on the *CAR\_PC* to obtain a single reference that is accurate to within a few nanoseconds [Lewandowski and Thomas \(1991\)](#). First, the GPS position for each LGPR frame is recorded based on the GPS device incorporated within the LGPR sensor. Next, the GPS sensor within the vehicle records a data-stream of pairs of time stamps ( $t_{sys}, t_{gps}$ ) at 100 Hz. Lastly, for each vehicle sensor

(GPS, odometry, camera, and lidar), each data point is recorded with the corresponding system time  $s_{sys}$ .

Then, to synchronize all of the onboard sensor data in post-processing, we interpolate the GPS reference time for each sensor data point as:

$$s_{gps} = \frac{s_{sys} - t_{sys}^i}{t_{sys}^{i+1} - t_{sys}^i} (t_{gps}^{i+1} - t_{gps}^i) \quad (3)$$

where  $(t_{sys}^i, t_{gps}^i)$  is the closest time reference pair with  $t_{sys}^i < s_{sys}$  and  $(t_{sys}^{i+1}, t_{gps}^{i+1})$  is the next consecutive time reference pair.

The data stream provided by the LGPR sensor contains geo-referenced radar frames. However, since the position estimates are obtained with the standard-quality GPS device integrated in the LGPR sensor, rather than the RTK-GPS system onboard the vehicle used for ground truth, we cannot simply match the LGPR frames to the ground truth positions using their recorded positions. Instead, to obtain accurate positions for LGPR frames, we first obtain timestamps by noting that the radar frames are collected at a fixed frequency of  $\approx 126$  Hz. Therefore, to obtain timestamps, the LGPR stream is aligned with the GPS data stream. We first differentiate the positions from both the LGPR scans and the RTK-GPS systems using a centered Savitzky-Golay filter [Savitzky and Golay \(1964\)](#). Next, we obtain the alignment offset between the velocities by maximizing the Pearson correlation coefficient between the velocities using numerical optimization. The result of this maximization, combined with the fixed frequency of the LGPR scans, allows every LGPR scan to be labeled with a corresponding timestamp on the GPS clock.

### Calibration

Each of the sensors included in the data stream was calibrated as needed before each run. Here we describe the various calibration processes for each sensor. Additionally, when possible we include the calibration files in the dataset (see Fig. 4).

**LGPR Sensor Calibration** The LGPR sensor needs to be calibrated to ensure changing environmental conditions do not unduly affect the sensor readings. The sensor includes a calibration routine which pulses each element for a short time and measures and records the mean values. This enables the intensity data to be stored in a mean-centered format, which helps remove any biases due to ambient conditions. This allows the device to operate reliably in temperatures ranging from  $-5^{\circ}\text{C}$  to  $50^{\circ}\text{C}$  [Cornick et al. \(2016\)](#). We calibrated the LGPR sensor at the start of each day. Additionally, in

the event the temperature changed drastically throughout the day, the calibration routine was repeated in between runs. For maximum flexibility, we include for each frame both the raw LGPR frame as *frame\_id.gpr*, and the mean-centered frame as *frame\_id.gmr*. The transform between the vehicle frame and LGPR frame is provided in *vehicle\_to\_lgpr.yaml* (see Fig. 4).

**RTK-GPS Calibration** The RTK-GPS system requires calibration in two ways. Firstly, because the GPS is receiving base station corrections, all measurements are with respect to the fixed location of the base station. Any errors in the measurement of the location of the base station itself would propagate to the vehicle measurements and reduce the system accuracy. To mitigate this, we fix the location of the base station by mounting it in a permanent position. For each of the road types *rural* and *urban* we selected a fixed base station antenna mount and recorded and averaged GPS positions for one hour. Once that averaging period was complete, we record the mean position and use it for all future runs using that base station position. This ensures that no errors are introduced between runs due to incorrect measurement of the base station position. Note that for *road\_type* highway we do not use corrections because of the limited base station range.

The second calibration necessary for the GPS groundtruth is a built-in calibration routine in which the vehicle is driven in several loops and  $\infty$ -paths. The manufacturer provides a software tool for using these drives to fine-tune the position and orientation of the sensor within the vehicle, as well as the relative positions of each of the two roof-mounted GPS antennas. We store those values onboard the device, and provide them with the dataset in *vehicle\_to\_gps.yaml* (see Fig. 4).

**Camera Calibration** To calibrate the Point Grey Grasshopper front-facing camera, we utilize the camera calibration method described here [ROS](#). The intrinsic calibration file is included in the dataset as *intrinsic/camera.yaml*. The measured extrinsic calibration between the camera and the velodyne is provided in *velodyne\_to\_camera.yaml*, as shown in Fig. 4.

**Velodyne Calibration** The Velodyne sensor includes a factory-provided calibration file that accounts for small differences in manufacturing and assembly and the effect they have on the conversion between the measured azimuth/angle of each point, and its position in the sensor frame. We apply this calibration file in order to produce the Velodyne frames found in *frame\_id.pcd*. We also include the

calibration file in *intrinsic/velodyne.xml*. Additionally, we provide the transform between the Velodyne sensor frame and the vehicle frame in *vehicle\_to\_velodyne.yaml*, as shown in Fig. 4.

## LGPRNet: Learning to Localize in Weather

Autonomous navigation in inclement weather was demonstrated in [Ort et al. \(2020\)](#). There, a degradation in the quality of the localization was found in inclement weather, in particular for rain. Previously, we described a benchmark to compare algorithms that address this issue. Here, we demonstrate an application of using the GROUNDED dataset to train a Convolutional Neural Network (CNN) to localize in a variety of weather conditions using a network architecture we call LGPRNet.

### Problem Formulation

To address the Localization in Weather benchmark, we define the localization problem comprising a prior map  $M^W$  in the world frame consisting of a set of  $N$  LGPR frames  $F_i$  for  $i \in \{1..N\}$  where each frame  $F_i \in \mathbb{R}^{N_D \times N_C}$  and  $N_D$  is the number of depth bins and  $N_C$  is the number of channels in the radar array. Thus, the value  $F_{i,d,c}$  represents the intensity of the radar return in the  $i$ th frame at a particular depth and channel. Furthermore, each frame  $F_i$  is associated with a pose  $P_i$  which contains the pose of the sensor in the world frame at the time the frame was measured. Next, we have a target frame  $F_T \in \mathbb{R}^{N_D \times N_C}$  which is measured when the vehicle revisits the mapped region. Our goal is to find the target pose  $P_T$  which gives the pose of the sensor at the time  $F_T$  was measured.

We simplify the problem by noting that even coarse GPS estimates enable focusing on a Region of Interest (ROI) which can significantly narrow the size of the map that must be searched. To this end, we replace the full map  $M^W$  with a target specific window  $W_T$  of length  $N_W$ , a fixed hyper-parameter, which selects linearly-spaced frames from  $M^W$  in the region of  $P_C$ , a coarse estimate of the target pose  $P_T$ . Since the LGPR collects frames at a fixed frequency, the spatial density of the map frames varies by the speed of the mapping vehicle. Therefore, the spatial size of the window is variable, and can depend on the uncertainty in the coarse pose estimate.

### LGPRNet Architecture

We propose a network architecture for LGPR localization, LGPRNet, based on the Inception architecture [Szegedy et al. \(2015\)](#) which excels at detecting features at varying

scales and uses dimensionality reduction to allow for a very deep architecture while reducing the computational cost. The problem of detecting underground radar features is quite distinct from the ImageNet Deng et al. (2009) object detection task, for which pre-trained models are available. Additionally, the dimensionality of the input features is unique in this problem because the number of channels in a radar sensor is quite low compared to the number of depth bins, creating a very narrow aspect ratio (33:1), while the number of frames in each window region can be larger than the typical 3-channel, low-aspect-ratio RGB images used as input in those networks. For these reasons, we design a custom network architecture (see Fig. 7), where the layer parameters and kernel shapes are specifically designed for the unique LGPR feature shapes, and train it using GROUNDED data.

The input features shape is  $(B, N_f, N_c, N_D)$ , where  $B$  is the batch size,  $N_f = N_w + 1$  is the number of frames in the input including the frames from the window and the target,  $N_c$  is the number of radar channels, and  $N_D$  is the number of depth bins in each frame. Note that while the batch size is variable,  $N_f, N_c, N_d$ , are fixed for a given trained model. For our implementation  $N_f = 10, N_c = 11, N_D = 369$ . Due to the unusually large aspect ratio (33:1), we design rectangular kernels such as (3,11) in the first convolution, and (3,7) in the second, as well as rectangular max pooling windows such as (2,3) in the first MaxPool, which balances the sparsity in the channel dimension with the density in the depth dimension. Finally, after the last fully connected layer in each of the three outputs (Main, Aux1, and Aux2) we add a sigmoid activation to transform to the output range of each pose dimension.

We choose an output representation which is relative to the pose of the input window. This obviates the need to include the real-world pose as an input and simplifies training. Instead, the predicted pose is a 3-tuple of [longitudinal, lateral, rotation], where longitudinal is a float in [0,1] which represents the longitudinal position with respect to the input window. The lateral position is a float in [0,1] which indicates the lateral offset from the window with respect to the sensor width. Since the sensor must be overlapping at least partially with the map in order to localize, the range 0,1 represents the full range of [-50, 50)% overlap in the lateral direction. Finally, the rotation is the angle in radians, normalized to the distribution in the dataset, where 0 indicates the target pose is aligned with the center of the map window. In this way, the model can predict an output pose relative only to the window, and the final vehicle pose prediction is then computed using the known window frame poses.

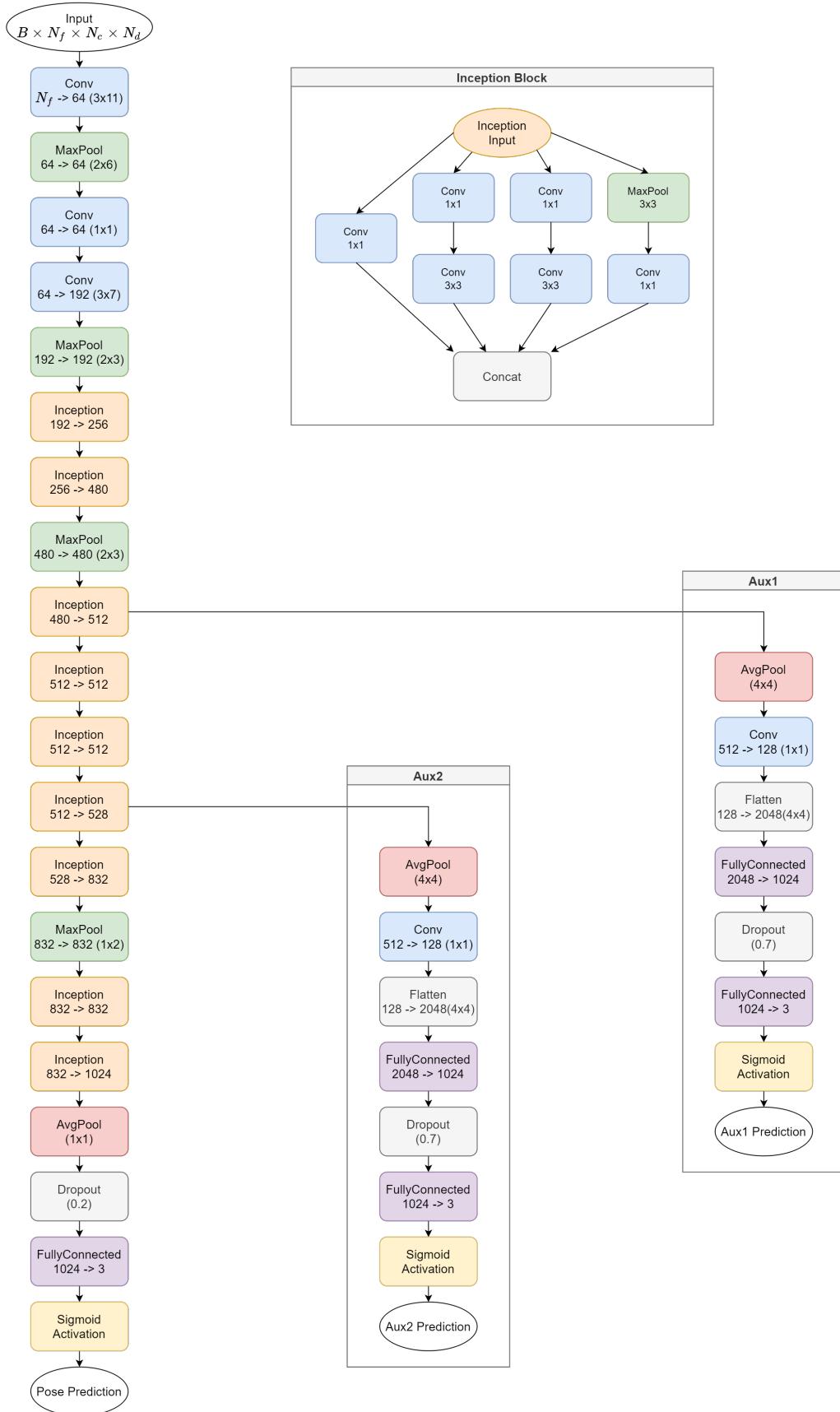
Using window-relative output poses in this way simplifies training. However, it does impose a limitation on the possible predictions of the network. For example, the prediction will always lie within the input window longitudinally, and will always assume at least 50% overlap with the map frame laterally. Therefore, it is important to choose a large enough window to capture the uncertainty in the initial pose estimate. It would be interesting to also predict an output reflecting the likelihood that the window *doesn't* contain the target scan; however, we leave that for future work.

## Evaluation

**Training** We train the LGPRNet on 72 runs from the GROUNDED dataset. Since the goal of this benchmark is to localize in inclement weather conditions, we choose a selection of map/target pairs that include different weather conditions between the mapping and localization. (Please see [Supplementary Material](#) for the complete list.)

Each map/target pair consists of two runs collected on the same geographic route. To create labeled features, we use the GPS measurements provided in the dataset. Since the GPS measurements are provided at 100 Hz while the LGPR measurements are provided at 126 Hz we use linear interpolation between the GPS measurements to obtain a groundtruth pose at each LGPR frame. Note that extra care must be given to angular measurements to ensure interpolation properly wraps around  $\pm\pi$ . Next, for each target scan, a set of training examples is created by selecting a set of possible map window regions that contain the target scan. Since each target can be located anywhere in the window, and the windows can vary in size, there is a combinatoric number of training examples that can be obtained from each map/target pair. For example, in a 5-minute drive containing  $\approx 50k$  radar frames,  $\approx 100M$  training examples can be generated. While this is more than enough data for training the network, using only data from a single run would quickly overfit to that location. Instead, we randomly subsampled 100k samples from each of the 72 runs to obtain a training set of 7.2M labeled examples. for training.

We further hold back 100k randomly selected examples for validation and train the network on a single Nvidia 1080Ti GPU until convergence (approx 24 hours). During training, we perform back-propagation at both the main output layer as well as two auxiliary outputs, which aids with training the earlier layers. We weighted these losses 50% on the output and 25% on each of the auxiliary heads. We also apply a loss weight on the regressed pose outputs to account



**Figure 7.** LGPRNet architecture composed of traditional convolutional layers along with Inception building blocks includes a total of 20 layers (Not including the auxiliary outputs, and considering each inception block a single layer). Notice that many of the kernels are not square to account for the unusually high aspect ratio of LGPR frames.

for the different units between the translation and rotation components, as described in [Localization in Weather Metric](#).

**Results** To evaluate the trained LGPRNet model, we select 10 distinct map/target run pairs, none of which were included in the 72 runs used for training. Additionally, we ensured that there was no geographic overlap between any of the evaluation runs and the training runs. This ensures that the model is learning the relative pose between the target and the map and not learning the map itself. To avoid requiring LGPR maps to be built specifically for a particular weather condition, we only use maps collected in clear weather for the Localization in Weather benchmark. In this evaluation, we compare the localization quality when the target is in a variety of weather conditions including clear weather, snow, and rain, while localizing to the same clear-weather map. (Please see [Supplementary Material](#) for the complete list of runs used for evaluation.)

		Clear	Snow	Rain
Mean Error	Correlation	0.34	0.39	0.77
	LGPRNet (Ours)	0.32	0.44	0.47
Lateral	Correlation	0.26	0.29	0.40
	LGPRNet (Ours)	0.16	0.26	0.26
Longitudinal	Correlation	0.17	0.21	0.57
	LGPRNet (Ours)	0.24	0.30	0.33

**Table 5.** The results of the LGPRNet predictions compared the hand-designed optimal correlation algorithm used in [Ort et al. \(2020\)](#). The mean distance error (m) is compared as well as its components in the lateral and longitudinal directions with respect to the lane direction.

Table 5 shows the evaluation results compared with the results from [Ort et al. \(2020\)](#), which used a heuristic correlation-based optimization to localize. For a hypothetical pose and a given scan in the map, the correlation is calculated as

$$r_{A,B} = \frac{\sum_{i,d} A_{i,d} B_{i,d}}{\sqrt{\sum_{i,d} A_{i,d}^2 B_{i,d}^2}} \quad (4)$$

where  $A$ , and  $B$  represent the current scan, and the scan data from the prior map respectively,  $i$  spans the number of channels, and  $d$  spans the number of depth bins. The optimal match that maximizes  $r_{A,B}$  in (4) is then used as the localization estimate for the correlation baseline (see [Ort et al. \(2020\)](#) for more details).

We see in Table 5 that the learned model compares similarly for clear weather and snow, while significantly improving on the localization quality in rain. Importantly, the metrics reported for the correlation-based approach were

considered for localization in each weather, to various maps including both matching and non-matching weather maps. Here, in keeping with the specification for the benchmark, we evaluate rain and snow conditions using only maps collected in clear weather, which is a more challenging task.

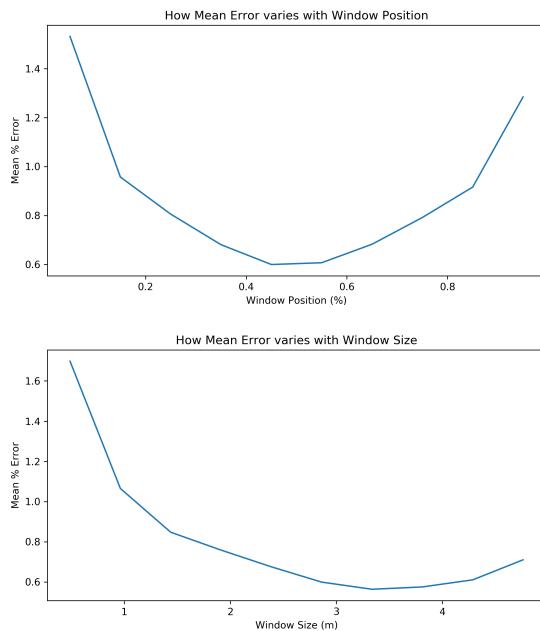
We interpret the degradation in localization quality in rain as being due to the changing dielectric properties of the soil when it is saturated. This could impact the radar signal by effectively scaling the entire frame along the depth dimension. Thus, while the previous correlation-based approach could not account for this variation in scale, LGPRNet learns to correct for this through the inclusion of varying weather conditions in the training set. On the other hand, the lower error found in the snowy weather conditions is likely not caused by a systemic change in the ground dielectric content, since the snow sits mostly above the ground and doesn't saturate it like rain does. Instead, this layer of snow between the sensor and the ground could introduce some signal attenuation, which would appear as sensor noise. Similarly, the error in clear weather localization is likely due in part to sensor noise, as well as imperfect overlap between the target and map data. One possible solution that could refine this estimate in practice would be to fuse these absolute measurements with a proprioceptive state estimation system including an IMU and/or wheel encoders, as was done in [Ort et al. \(2020\)](#). However, here we focus on evaluation of the LGPRNet prediction directly.

Also important to note is that for autonomous vehicles, error in the longitudinal (along lane) direction is often less critical than the lateral direction (across lanes). Table 5 shows that the LGPRNet performs particularly well in the lateral direction, where it has a maximum of 26 cm of deviation even in rain and outperforms the correlation approach in all weather types. This is accounted for by the weights in the Localization in Weather benchmark described previously. Using Eq. (1) we compute benchmark values of:

$$s_{\text{snow}} = 0.585$$

$$s_{\text{rain}} = 0.595$$

Finally, we also investigate how the position of the target in the window, and the size of the search window affect the localization accuracy. Fig. 8 shows that the optimal position for the target is in the center of the window, as the error increases by about 1% at each end. This is likely due to the center of the window containing the smallest average distance to the frames in the window. Interestingly, with respect to window size, there appears to be an optimal size at



**Figure 8.** The localization accuracy is affected by the target position in the window (Top) as well as the size of the search window (Bottom).

approximately 3.5 m. This is likely a factor of the size of the features in the signal, as very small windows do not contain enough variation, while very large windows induce error due to the larger search space. Therefore, these data indicate that the optimal window size should be the minimum of 3.5 m and the uncertainty of the coarse pose estimate.

## Conclusion

In this work, we presented a novel dataset for localization and mapping research in autonomous driving using a ground penetrating radar. Our goal is to unlock the potential of radiogeological navigation, as this sensing modality holds the promise to enable autonomous vehicle localization even in the most challenging weather conditions. Together with the dataset, this work proposes evaluation protocols and additionally provides camera and lidar data to simplify comparisons with established algorithms on these sensing modalities.

There are also several limitations to this work which should be noted. Firstly, for practical reasons, the data is limited to urban, rural, and highway environments in Massachusetts, USA. We hope to be able to expand the dataset to a wider variety of locations in the future. To this end, arranging an alternative groundtruth method such as satellite corrections rather than a fixed base station would greatly improve the range for the data collection.

Next, additional sensor fusion could be utilized both to improve the ground truth localization and as a baseline for

lidar and/or camera-based methods. While the reader could implement this using the sensor streams from the dataset, these results may also be added to the dataset to improve usability.

Finally, we provide time synchronization between the various sensors. However, these timestamps are based on system time recorded when a data point is received from each sensor, rather than a trigger used to ensure that all sensor readings are synchronized which would be more precise.

## Acknowledgements

This work was supported in part by the MIT Lincoln Laboratory and ONR grant N00014-18-1-2830. The authors would also like to thank Mike Boulet and Byron Stanley for their continued advice and technical support throughout this project.

## References

- Baikovitz A, Sodhi P, Dille M and Kaess M (2021) [Ground encoding: Learned factor graph-based models for localizing ground penetrating radar](#). In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 5476–5483.
- Barnes D, Gadd M, Murcett P, Newman P and Posner I (2020) [The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset](#). In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 6433–6438.
- Benedetto A, Tosti F, Ciampoli LB and D’amico F (2017) [An overview of ground-penetrating radar signal processing techniques for road inspections](#). *Signal processing* 132: 201–209.
- Bescos B, Fácil JM, Civera J and Neira J (2018) [DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes](#). *IEEE Robotics and Automation Letters* 3(4): 4076–4083.
- Binas J, Neil D, Liu SC and Delbruck T (2017) [DDD17: End-to-end DAVIS driving dataset](#). *arXiv preprint arXiv:1711.01458*.
- Bürki M, Cadena C, Gilitschenski I, Siegwart R and Nieto J (2019) [Appearance-based landmark selection for visual localization](#). *Journal of Field Robotics* 36(6): 1041–1073.
- 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).
- Cadena C, Carlone L, Carrillo H, Latif Y, Scaramuzza D, Neira J, Reid I and Leonard JJ (2016) [Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age](#). *IEEE Transactions on Robotics* 32(6).

- Caeser H, Bankiti V, Lang AH, Vora S, Lioung VE, Xu Q, Krishnan A, Pan Y, Baldan G and Beijbom O (2020) [nuscenes: A multimodal dataset for autonomous driving](#). In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 11621–11631.
- Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S and Schiele B (2016) [The cityscapes dataset for semantic urban scene understanding](#). In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 3213–3223.
- Cornick M, Koechling J, Stanley B and Zhang B (2016) [Localizing Ground Penetrating RADAR: A Step Toward Robust Autonomous Ground Vehicle Localization](#). *JFR* 33(1): 82–102.
- Dawson-Howe KM and Williams TG (1998) [The detection of buried landmines using probing robots](#). *RAS*.
- Deng J, Dong W, Socher R, Li LJ, Li K and Fei-Fei L (2009) [Imagenet: A large-scale hierarchical image database](#). In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255.
- Doan AD, Latif Y, Chin TJ, Liu Y, Do TT and Reid I (2019) [Scalable place recognition under appearance change for autonomous driving](#). In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 9319–9328.
- Dymczyk M, Stumm E, Nieto J, Siegwart R and Gilitschenski I (2016) [Will it last? Learning stable features for long-term visual localization](#). In: *2016 Fourth International Conference on 3D Vision (3DV)*. IEEE, pp. 572–581.
- Fehr M, Furrer F, Dryanovski I, Sturm J, Gilitschenski I, Siegwart R and Cadena C (2017) [TSDF-based change detection for consistent long-term dense reconstruction and dynamic object discovery](#). In: *2017 IEEE International Conference on Robotics and automation (ICRA)*. IEEE, pp. 5237–5244.
- Geiger A, Lenz P, Stiller C and Urtasun R (2013) [Vision meets robotics: The KITTI dataset](#). *The International Journal of Robotics Research* 32(11).
- Huang X, Cheng X, Geng Q, Cao B, Zhou D, Wang P, Lin Y and Yang R (2018) [The ApolloScape Dataset for Autonomous Driving](#). In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. pp. 954–960.
- Kendall A, Grimes M and Cipolla R (2015) [Posenet: A convolutional network for real-time 6-dof camera relocalization](#). In: *Proceedings of the IEEE international conference on computer vision*. pp. 2938–2946.
- Kubota T, Ushio T, Shige S, Kida S, Kachi M and Okamoto K (2009) [Verification of High-Resolution Satellite-Based Rainfall Estimates around Japan Using a Gauge-Calibrated Ground-Radar Dataset](#). *Journal of the Meteorological Society of Japan*.
- Ser. II* 87A. DOI:10.2151/jmsj.87A.203.
- Levinson J, Montemerlo M and Thrun S (2007) [Map-based precision vehicle localization in urban environments](#). In: *Robotics: science and systems*, volume 4. Citeseer, p. 1.
- Levinson J and Thrun S (2010) [Robust vehicle localization in urban environments using probabilistic maps](#). In: *2010 IEEE international conference on robotics and automation*. IEEE, pp. 4372–4378.
- Lewandowski W and Thomas C (1991) [GPS time transfer](#). *Proceedings of the IEEE* 79(7): 991–1000.
- Liu K, Li Q and Qiu G (2020) [PoseGAN: A pose-to-image translation framework for camera localization](#). *ISPRS Journal of Photogrammetry and Remote Sensing* 166: 308–315.
- Lowry S, Sünderhauf N, Newman P, Leonard JJ, Cox D, Corke P and Milford MJ (2016) [Visual Place Recognition: A Survey](#). *IEEE Transactions on Robotics* 32(1).
- Maddern W, Pascoe G, Linegar C and Newman P (2017) [1 Year, 1000 Km: The Oxford RobotCar Dataset](#). *The International Journal of Robotics Research* 36(1): 3–15.
- Meyer M and Kuschk G (2019) [Automotive radar dataset for deep learning based 3d object detection](#). In: *2019 16th european radar conference (EuRAD)*. IEEE, pp. 129–132.
- 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.
- 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.
- Pitropov M, Garcia DE, Rebello J, Smart M, Wang C, Czarnecki K and Waslander S (2020) [Canadian Adverse Driving Conditions Dataset](#). *The International Journal of Robotics Research*.
- Rapp M, Hahn M, Thom M, Dickmann J and Dietmayer K (2015) [Semi-markov process based localization using radar in dynamic environments](#). In: *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, pp. 423–429.
- Romero-Ruiz A, Linde N, Keller T and Or D (2018) [A Review of Geophysical Methods for Soil Structure Characterization](#). *Reviews of Geophysics* 56(4): 672–697.
- ROS (2021) [Camera Calibration - ROS Wiki](#). URL [http://wiki.ros.org/camera\\_calibration#Camera\\_Calibrator](http://wiki.ros.org/camera_calibration#Camera_Calibrator).
- Rusu RB and Cousins S (2011) [3d is here: Point cloud library \(pcl\)](#). In: *2011 IEEE international conference on robotics and automation*. IEEE, pp. 1–4.
- Ryan D (2017) [Lincoln Laboratory enters licensing agreement to produce its localizing ground-penetrating radar](#). [https:](https://)

- //news.mit.edu/2017/lincoln-laboratory-enters-licensing-agreement-to-produce-localizing-ground-penetrating-radar-0718.
- Savitzky A and Golay MJ (1964) Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry* 36(8): 1627–1639.
- Schulz Y, Mattar AK, Hehn T and Kooij J (2021) Hearing What You Cannot See: Acoustic Vehicle Detection Around Corners. *Robotics and Automation Letters*.
- Shaw K (2020) Meet WaveSense: Providing Accuracy for Vehicles Via Underground Radar. <https://www.roboticsbusinessreview.com/unmanned/unmanned-ground/meet-wavesense-providing-accuracy-for-vehicles-via-underground-radar>.
- Sun P, Kretzschmar H, Dotiwalla X, Chouard A, Patnaik V, Tsui P, Guo J, Zhou Y, Chai Y, Caine B, Vasudevan V, Han W, Ngiam J, Zhao H, Timofeev A, Ettinger S, Krivokon M, Gao A, Joshi A, Zhao S, Cheng S, Zhang Y, Shlens J, Chen Z and Anguelov D (2020) Scalability in perception for autonomous driving: Waymo open dataset. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 2446–2454.
- Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V and Rabinovich A (2015) Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 1–9.
- Werber K, Klappstein J, Dickmann J and Waldschmidt C (2016) Point group associations for radar-based vehicle self-localization. In: *2016 19th International Conference on Information Fusion (FUSION)*. IEEE, pp. 1638–1646.
- Werber K, Klappstein J, Dickmann J and Waldschmidt C (2019) Association of straight radar landmarks for vehicle self-localization. In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, pp. 736–743.
- Williams RM, Ray LE and Lever J (2012) An autonomous robotic platform for ground penetrating radar surveys. In: *2012 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, pp. 3174–3177.
- Wing MG, Eklund A and Kellogg LD (2005) Consumer-grade global positioning system (GPS) accuracy and reliability. *Journal of forestry* 103(4): 169–173.
- Wolcott RW and Eustice RM (2014) Visual localization within lidar maps for automated urban driving. In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 176–183.
- Wolcott RW and Eustice RM (2015) Fast LIDAR localization using multiresolution Gaussian mixture maps. In: *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, pp. 2814–2821.
- Zhang Z and Scaramuzza D (2018) A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry. In: *(IROS)*. IEEE, pp. 7244–7251.
- Zhu AZ, Thakur D, Özaslan T, Pfrommer B, Kumar V and Daniilidis K (2018) The Multivehicle Stereo Event Camera Dataset: An Event Camera Dataset for 3D Perception. *Robotics and Automation Letters* 3(3).

## Supplementary Material

Data Split	Map ID	Run ID	Map Weather	Run Weather		Data Split	Map ID	Run ID	Map Weather	Run Weather
Train	6	44	snow	clear		Train	48	83	clear	rain
Train	7	44	snow	clear		Train	49	82	clear	rain
Train	6	45	snow	clear		Train	49	83	clear	rain
Train	7	45	snow	clear		Train	82	48	rain	clear
Train	6	78	snow	rain		Train	83	48	rain	clear
Train	7	78	snow	rain		Train	82	49	rain	clear
Train	6	79	snow	rain		Train	83	49	rain	clear
Train	7	79	snow	rain		Train	20	58	snow	clear
Train	44	6	clear	snow		Train	20	59	snow	clear
Train	44	7	clear	snow		Train	20	92	snow	rain
Train	45	6	clear	snow		Train	20	93	snow	rain
Train	45	7	clear	snow		Train	21	58	snow	clear
Train	78	6	rain	snow		Train	21	59	snow	clear
Train	78	7	rain	snow		Train	21	92	snow	rain
Train	79	6	rain	snow		Train	21	93	snow	rain
Train	79	7	rain	snow		Train	58	20	clear	snow
Train	44	78	clear	rain		Train	59	20	clear	snow
Train	44	79	clear	rain		Train	92	20	rain	snow
Train	45	78	clear	rain		Train	93	20	rain	snow
Train	45	79	clear	rain		Train	58	21	clear	snow
Train	78	44	rain	clear		Train	59	21	clear	snow
Train	79	44	rain	clear		Train	92	21	rain	snow
Train	78	45	rain	clear		Train	93	21	rain	snow
Train	79	45	rain	clear		Train	58	92	clear	rain
Train	10	48	snow	clear		Train	59	92	clear	rain
Train	10	49	snow	clear		Train	92	58	rain	clear
Train	10	82	snow	rain		Train	93	58	rain	clear
Train	10	83	snow	rain		Train	58	93	clear	rain
Train	11	48	snow	clear		Train	59	93	clear	rain
Train	11	49	snow	clear		Train	92	59	rain	clear
Train	11	82	snow	rain		Train	93	59	rain	clear
Train	11	83	snow	rain		Val	52	53	clear	clear
Train	48	10	clear	snow		Val	52	14	clear	snow
Train	49	10	clear	snow		Val	52	15	clear	snow
Train	82	10	rain	snow		Val	52	86	clear	rain
Train	83	10	rain	snow		Val	52	87	clear	rain
Train	48	11	clear	snow		Val	62	63	clear	clear
Train	49	11	clear	snow		Val	62	24	clear	snow
Train	82	11	rain	snow		Val	62	25	clear	snow
Train	83	11	rain	snow		Val	62	96	clear	rain
Train	48	82	clear	rain		Val	62	97	clear	rain

**Table 6.** Listing of map/target run pairs from GROUNDED used for training and evaluation of the LGPRNet model.