Low-Cost Urban Localization with Magnetometer and LoRa Technology

Derek Benham, Ashton Palacios, Philip Lundrigan, Joshua G. Mangelson

***Abstract*— With the goal of developing low-cost and innovative perception and localization techniques for autonomous vehicles, this work explores a system that solely relies on a LoRa receiver and a magnetometer for agent localization within urban environments. Using the received signal strength from LoRa beacons distributed across a test area of 16,000 square meters, a model of expected RSSI values per beacon is estimated using Gaussian Process (GP) regression. Motion is estimated using a probabilistic signal similarity classifier, and localization is obtained via a particle filter. Our experiments demonstrate that our proposed system is able to estimate our location to within three meters RMSE when the agent is within the convex hull of prior data. In real-world scenarios, characterized by signal interference and environmental complexities, our approach highlights the potential of leveraging affordable technology such as LoRa receivers and magnetometers for robust and accurate location estimation in complex urban environments. The integration of low-cost LoRa devices, Gaussian Process regression, particle filtering and our novel signal similarity motion estimator offers a promising avenue for achieving cost-effective localization solutions without compromising accuracy or reliability.**

I. Introduction

Many approaches to localize autonomous vehicles include the use of cameras, LiDAR, and GPS sensors. These sensors, albeit very common, vary in accuracy depending on cost and often require an extensive prior map. GPS, on the other hand, offers a direct global positioning solution, but accuracy is compromised in urban landscapes due to obstructed satellite views, multipath signal propagation, and signal fading. This phenomenon is particularly pronounced in densely populated areas or urban canyons where GPS accuracy is often severely degraded.

In an effort to address these challenges, state estimator solutions typically fuse odometry with multiple additional sensors to obtain robust position estimates. Encoders are a popular method for estimating odometry for wheeled robots. By measuring rotational motion directly from the wheels, encoders provide accurate pose estimation over short intervals. However, their utility is limited to wheeled robots, relying on well-defined wheel parameters and consistent, smooth movement. This constraint excludes other mobile robot types such as legged, aerial or marine platforms, as well as handheld devices, which lack the necessary kinematic characteristics for encoder-based odometry.

Besides GPS, other wireless signals are naturally present in urban environments that can be used to localize, such as cellular signals or WiFi. Another wireless technology that is increasing in popularity in the United States and is well established in Europe is LoRa. For localization, LoRa provides unique advantages compared to cellular signals or WiFi due to it being low cost, low power, and long range. In contrast, cellular networks often face challenges with insufficient visible cell towers and limited resolution in signal strength. WiFi on the otherhand, present challenges due to its limited range and high power consumption.

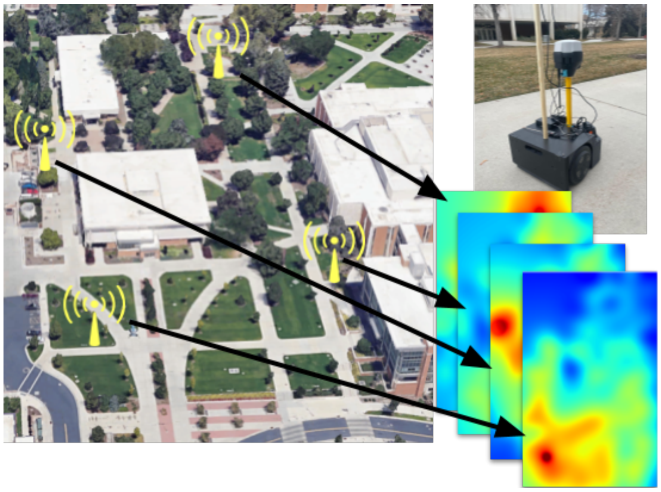

Fig. 1. Our experimental low-cost research platform attempts to localize via received signal strength values from four different LoRa beacons on campus. The signal strength of each beacon is individually modeled by a GP. Motion is estimated using a magnetometer and a novel wireless signal strength similarity classifier. These motion estimates are then fused with observations of LoRa signal strength measurements using a particle filter.

In this work we localize an agent in an urban environment solely using the signal strength of LoRa and a magnetometer, see Fig. 1. We employ Gaussian Process (GP) regression to model a prior signal strength heat map (III), utilize a probabilistic motion classifier for motion estimation (IV), and accomplish localization through sequential Monte-Carlo sampling (V). Experiments show improvement over existing methods (VI).

The contributions of this paper include:

- To our knowledge, the first use of commodity off-the-shelf (COTS) LoRa technology for mobile agent localization,
- A novel signal strength similarity motion classifier using Gaussian Mixture Models, and
- A robust localization system utilizing LoRa received signal strength and a magnetometer.

II. Related Work

As more wireless devices come online, there has been a renewed focus in using the wireless signals to localize agents

D. Benham, A. Palacios, P. Lundrigan, J. Mangelson, are at Brigham Young University.
{laser14, apal6981, lundrigan, mangelson}@byu.edu.

in an environment. Whether indoors, in urban canyons or on low cost, low power devices, GPS can not always be relied upon to provide accurate position estimates.

A. Signal Strength Based Localization

The simplest implementation of signal-strength-based localization is trilateration where signal strength is mapped to a distance between a transmitter and a receiver. To increase trilateration accuracy, a weighted least squares approach was proposed in [1], [2]. By assigning weights to signal strength measurements based on estimated reliability, the method proposed in [1] reduces the influence of noisy or unreliable data. Such a technique can be sufficient in open environments, but can struggle in cluttered urban environments as it does not attempt to model the aforementioned signal multipath issues.

A common approach to localization in non-linear environments is fingerprinting, a process in which a unique signature or “fingerprint” is created for a specific location based on the characteristics of wireless signals [3]. This is done through the use of known correspondences to access points. In WiFi these known correspondences are the MAC addresses of the router. The most common fingerprint is the received signal strength indicator (RSSI) which is representative of the power present in a received radio signal, a measurement built into most wireless chipsets. To localize via fingerprinting, a reference database is created by associating measurements with known locations. By matching real-time signal measurements to the database, the system estimates the user’s location. A drawback to many fingerprinting techniques is the necessity of a high fidelity, well covered map.

In an attempt to infer values missing in a fingerprinted map, [4] uses GP regression. GP based techniques enable estimating the likelihood of a given measurement because a GP models both mean and uncertainty of an observed signal.

Early attempts at combining a particle filter with a prior map was done by [5] where a GP regression estimator was used in an outdoor environment attempting to localize using the RSSI values from cellular towers. Following 208 hours of driving through downtown Seattle and its suburbs to gather training data, their model was shown to estimate their relative position within the range of 130 to 230 meters.

Instead of directly observing position, the authors in [6] attempt to estimate a robot’s motion using differential signal strength measurements from static WiFi access points in unknown locations. They estimate the magnitude of translation of a robot between time steps using a multilayer perceptron. A magnetometer is then used to determine heading. The authors employ various similarity score metrics [7], [8] to extract features from signal strength measurements. At each time step t, similarity scores are computed by comparing RSSI measurements with the preceding time steps from $t-1$ to $t-w$, where w represents a user-defined window. With these inputs, their model predicts a magnitude travelled by the robot with heading estimated by a magnetometer. Their method achieves position estimates within one to six meters of error through dead reckoning, outperforming wheel odometry. Through data exclusion testing, the authors found that their method still required a connection to a significant number of access points.

B. LoRa Integration in Robotics

LoRa is a wireless communication technology designed for low-power, long-range communication in the Internet of Things (IoT) domain. Developed to address the specific requirements of IoT devices, LoRa enables devices to communicate over significant distances while consuming minimal power and minimal overhead cost. Most transceiver boards cost in the range of $10 USD. Unlike many WiFi or Bluetooth technologies, LoRa was designed to support battery powered embedded devices [9]. The technology operates in the sub-gigahertz frequency bands, providing an advantageous balance between range and energy efficiency. LoRa employs a unique spread spectrum modulation technique, allowing for robust communication in challenging environments with obstacles and interference. The technology has found applications in various sectors, including smart cities, agriculture, industrial automation, and environmental monitoring, contributing to the expansion and efficiency of IoT networks. With growing popularity in LoRa enabled devices, recent work has investigated the efficiency of localization techniques using LoRa, with a recent survey by [10]. Few have investigated robotics applications using LoRa.

Many implementations have used LoRa as a communication technology. Junaedy *et al.* [11] uses LoRa for inter-agent communication in a SLAM system and [12] uses it for teleoperation of mobile robots, sending all data to the operator in order to run a 2D-SLAM algorithm.

In [13], LoRa is utilized atop a mobile robot to track targets in an indoor facility, achieving a sub meter median error across their 100 m^2 lab space. Their approach however relies on modified hardware to extract angle of arrival.

A significant amount of work has been done in the IoT community to localize static, unmodified commercial hardware devices, such as remote sensors. Work done in [14] focused on localizing static LoRa devices to a prior map in a 37,500 square meter area. By modeling the signal strength of over 10 beacons with a GP, they were able to achieve accuracy of 20 to 30 meters RMSE. However, because we are interested in localizing a mobile robot, we additionally leverage the robot motion to assist in determining its location.

The authors of [15] believe their work to be the first integration of drones into the LoRa network. With the intent of aiding in search and rescue techniques by locating an object with a LoRa beacon, drones with LoRa enabled devices are deployed to improve upon initial device location as estimated by the network. The results of their field tests show that they can decrease location uncertainty from 300 meters to 30 meters.

LoRa based localization has the opportunity to provide a unique navigation solution as the broadcast range is greatly increased compared to traditional WiFi routers, while still maintaining differentiable signal strength in smaller environments unlike traditional cellular towers. LoRa’s low cost, low power ability to provide bidirectional communication makes

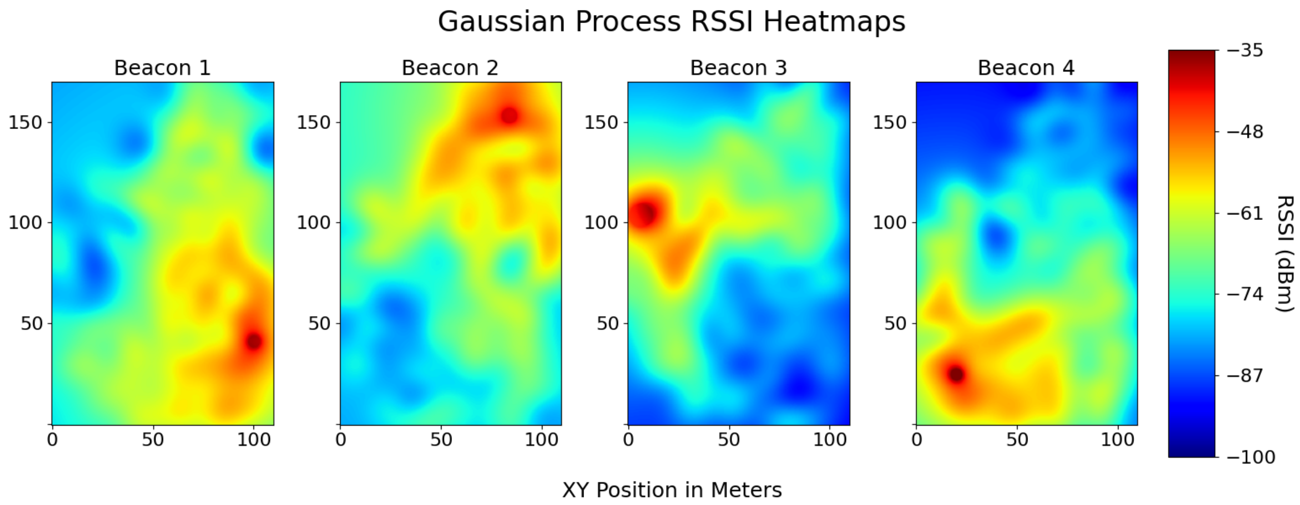

Fig. 2. Signal strength heat maps of each beacon across the environment, modeled using Gaussian Process Regression. An RBF kernel is employed with a non-uniform mean function to accurately represent signal decay in areas lacking data.

it a potentially ideal solution in swarm robotic localization applications. To the best of our knowledge, we are the first to develop a system for localization of a mobile robot using unmodified COTS LoRa hardware.

III. LoRa Signal Strength Modeling with Gaussian Processes

The challenge of nonlinear signal multipath propagation presents a significant obstacle in accurately modeling signal strength. Traditional methods often struggle to effectively capture these complexities, whereas GP regression offers a promising solution by inherently accounting for nonlinearities and uncertainties in spatial data modeling. In this section we explore how GPs can be used to effectively model the spatial distribution of LoRa signal strength, providing a robust framework to address for enhancing localization accuracy.

To generate our model we utilize GP regression [16], [17] to estimate a separate GP per beacon. The modeled heat maps can be seen in Fig. 2. A GP quantifies uncertainty by estimating a distribution over functions, enabling the assessment of uncertainty at each input point via the covariance structure. As such, GPs capture the relationships between data points and estimate the model confidence. We can view our gathered training points as measurements of the function $z_i = f(x_i) + \epsilon$ where x_i is an (x, y) tuple of our sampled location on campus, z_i is the recorded signal strength of the beacon represented as a negative dBm value, and ϵ is an additive zero-mean Gaussian noise with known variance σ^2. The GP estimates the posterior distribution over functions f from our training data. The underlying assumption of a GP is that function values are correlated in space according to a kernel function $k(\mathbf{x}_1, \mathbf{x}_2)$. In practice, the kernel function is user defined. We test several kernel functions including the Gaussian Radial Basis Function (RBF), Rational Quadratic and the Matérn32 kernel [18], our results are discussed in section VI.

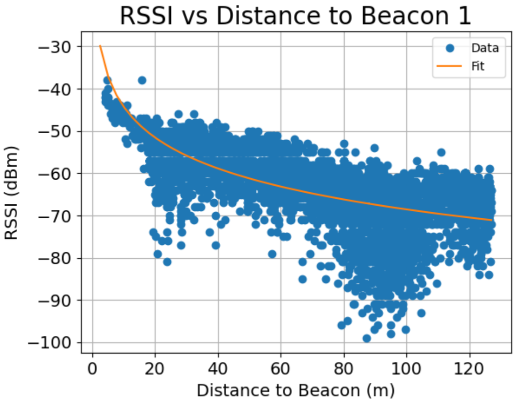

Fig. 3. Relationship between signal strength and the L^2 distance to beacon 1, revealing a logarithmic decay. To enhance our Gaussian Process's ability to infer in regions lacking data, a best-fit line, represented by $y = a + b\ln(x)$, is calculated through linear least squares fitting. The data exhibits significant noise due to the signal multipath and fading issue.

The RBF kernel is defined as

$$k_{\text{RBF}}(x, x') = \sigma^2 \exp\left(-\frac{(x - x')^2}{2\ell^2}\right) \quad (1)$$

with the lengthscale ℓ and the variance σ serving as the only parameters. The lengthscale ℓ governs the smoothness of the function and controls how rapidly the function changes over distance, while the variance σ dictates the average distance of the function from the mean. These hyper-parameters are tuned using gradient-based optimization within the framework of variational inference, which iteratively adjusts the variational parameters to minimize the difference between the approximated and true posterior distributions. We use the popular probabilistic programming package Pyro [19] for all our GP inference.

A. Mean Function Modeling

A mean function is provided to establish a prior, allowing a GP to make predictions where no training data exists. Typical

GPs often utilize a zero or constant mean function, which proves inadequate for modeling signal strength. A simplistic yet powerful way to model signal strength is with a simplified lognormal channel function such as $y = a + b\ln(x)$, where y is the expected signal strength, x is the L^2 distance from the transmitter, and both a and b are coefficients that need to be learned [20]. We use this function as our mean function because it accounts for the logarithmically decreasing signal pattern that isotropic antennas exhibit. We use least squares to solve for the coefficients of our training data using the following equations:

$$a = \frac{\sum_{i=1}^{n} y_i - b\sum_{i=1}^{n}(\ln x_i)}{n} \tag{2}$$

$$b = \frac{n\sum_{i=1}^{n}(y_i \ln x_i) - \sum_{i=1}^{n} y_i \sum_{i=1}^{n} \ln x_i}{n\sum_{i=1}^{n}(\ln x_i)^2 - (\sum_{i=1}^{n} \ln x_i)^2} \tag{3}$$

where y_i and x_i form a measurement pair relating signal strength to distance and n is the number of training points. By using a realistic mean function, our model best represents the true state of the data. Fig. 3 shows a line of best fit for the signal decay of beacon 1. The data is considerably noisy which is attributed to multipath propagation, signal fading, and loss of line of sight.

B. Sparse GPs

A drawback of GPs is their computational complexity, which scales to $O(n^3)$ when evaluating the covariance structure where n is the number measurements. With over 6000 training points collected from our vehicle, computing the uncertainty evaluation remains feasible but becomes slow, rendering it impractical for larger datasets. To account for this we employ the use of Sparse GPs (SGP) [21]. SGPs employ inducing points to approximate the complete Gaussian model, effectively reducing computational complexity. This reduction brings the complexity down to $O(mn^2)$, where m is the number of inducing points. To sub-sample our data we chose $m = 600$ initializing inducing points at the location of every 10th data point.

IV. Probabilistic Signal Similarity Motion Classifier

The overall aim of this work is to develop a localization system solely reliant on a magnetometer and a LoRa receiver. Achieving this requires the ability to infer motion without GPS or conventional wheel encoders. To address this challenge, we utilize the received signals from LoRa beacons to estimate odometry.

Expanding on the concepts introduced in the recent WiFi Similarity Odometry paper [6], which proposes a regression-based method for estimating an agent's motion using sequential signal strength measurements, we tailor our approach to suit our LoRa network constraints. While the authors of [6] demonstrated the functionality of their technique with as little as 20 access points, we encountered challenges when applying it to our smaller network of only four beacons. In our experiments, a perceptron system based on [6] often resorts to selecting the data mean or exhibited overfitting. Instead of relying on a regression technique to predict the agent's translation, we optimize the feature extraction methods and develop a probabilistic classifier to determine whether our agent has moved between time steps. Moreover, the developed classifier assigns a likelihood to each choice, which we utilize when sampling motion updates in a particle filter, detailed in section V.

A. Features

To extract features we compare RSSI values between time steps to create a similarity score. Since the association of beacons is known, and because LoRa signals are strong enough to be received throughout our testing area, we compare similarity scores for each beacon RSSI individually. To extract a similarity feature F for beacon n we use another form of the Gaussian Radial Basis function.

$$F_n(R_t, R_{t-i}) = \exp\left(-\frac{(R_t - R_{t-i})^2}{2\sigma^2}\right) \tag{4}$$

In this function, R_t and R_{t-i} indicate RSSI measurements for beacon n at time steps t and $t-i$. Measuring the variance of RSSI measurements, we find $\sigma = 1.5$. By combining these similarity scores together for each beacon, four features are extracted when evaluating R_t and R_{t-i}, creating a feature vector of $[F_1, F_2, F_3, F_4]_{t,t-i}$

To establish a history of these features, we implement a sliding window w, where each measurement is evaluated for a similarity score with the w previous measurements. Feature vectors are concatenated across a measurement window to create an input signal $\mathbf{F}$, where $\mathbf{F} = \{[F_1, F_2, F_3, F_4]_{t,t-1}, [F_1, F_2, F_3, F_4]_{t,t-2}, ..., [F_1, F_2, F_3, F_4]_{t,t-w}\}$. In our experiments, we find a sliding window of length $w = 3$ to be sufficient. At a few timesteps our agent did not receive measurements from all four beacons, to handle these scenarios we add the following checks. If a signal isn't received from beacon n during both time step t and $t - i$, we assign a similarity score of $F_n = 0.7$, indicating a moderately similar signal strength. If a signal is received at time step t but not $t - i$ (or at $t - i$ but not t) we assign $F_n = 0.3$, indicative of a weak signal similarity.

To train a classifier capable of distinguishing between movement and no movement, the robot would periodically stop for segments between 5 and 20 seconds while collecting data. This enables a collection of movement and non-movement examples from across the entire test area. The feature vectors $\mathbf{F}$ are then assigned a label. The classes of "movement" and "no movement" are determined based on whether the position of the robot between time step t and $t - 1$ differed by less than 0.15 meters. This threshold is selected according to the accuracy of our RTK GPS to establish ground truth. (The RTK has an accuracy of ±0.1 meters).

Due to the disproportionate number of instances labeled as "movement," we employ sub-sampling to ensure an even distribution of classes for training and evaluation of our model. Fig. 4 shows the distribution of feature vectors

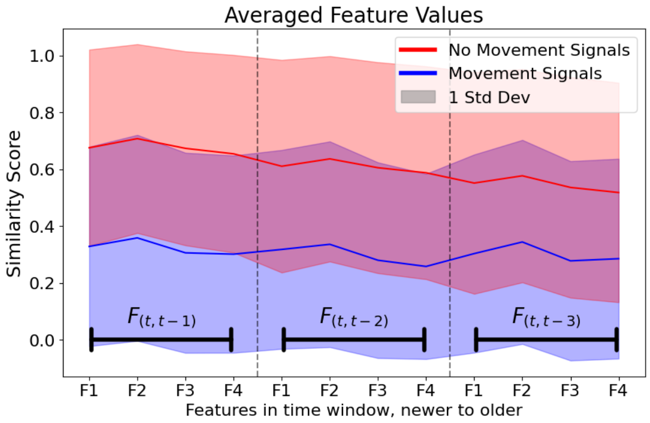

Fig. 4. Distribution of similarity scores for movement and no movement classes across all four beacons, shown with shaded regions indicating one standard deviation from the mean. With this amount of data overlap, denoted in purple, a simple binary classifier is insufficient for classification.

distinguished by label. Because of the significant overlap among the labels, a basic binary classification approach would prove inadequate for accurate classification.

B. Gaussian Mixture Model

An understanding of uncertainty is crucial for our model as mis-classifications can severely hinder performance. In lieu of employing a multi-layer perceptron or a k-nearest neighbor classifier, we classify the different movement labels using Gaussian Mixture models (GMM). Using a GMM allows a likelihood to be associated with each prediction. At training time, a separate GMM is fit for each class of “movement” and “no movement” features. During evaluation, each feature vector **F** is assessed by both GMMs, assigning a likelihood for each label. Total probability is calculated by adding together the likelihoods of each class, and the posterior is calculated by normalizing each likelihood by the total probability. Since the distribution of classes are evenly split, the prior probabilities are assumed to be equal.

When exclusively selecting the label with the higher posterior probability, our model selects the correct choice with approximately 87% accuracy. The confusion matrix of our model can be seen in Fig. 5. By leveraging the posterior likelihood to ensure that a class has a normalized posterior probability of at least 0.9, we observed an increase in model accuracy up to 95%. How we handle these decisions and leverage the likelihood is further discussed in Section V.

Because the input to our model is a sliding window, some of the labels of “no movement” include the scenario where the robot is in motion at time step $t - w$, but stopped at time t. In such situations there would be a dramatic change between the similarity scores between recent and older measurements. To ensure our model was not relying on these changes to infer decisions, we evaluated the model accuracy after pruning all feature vectors that contained a mixture of “movement” and “no movement” between time steps t and $t-w$. We concluded that our model was not over using these changes as the accuracy only dropped to 83%.

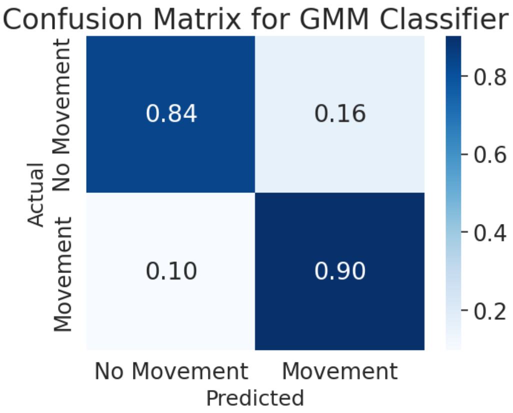

Fig. 5. Normalized confusion matrix of signal similarity Gaussian Mixture Model Classifier with approximately 87% accuracy on evaluation set. Both the training and test sets are evenly distributed across each class type to ensure accurate classification

V. Localization via Particle Filtering

We employ a Rao Blackwellized particle filter [22] with uniformly distributed starting particles to localize our vehicle within our test area. Our filter consists of the three typical particle filter stages: motion update, particle weight update, and resampling of particles. We preform these stages sequentially using the heading and RSSI values measured by the vehicle.

To determine the vehicle’s motion, we utilize our GMM classifier to inform an estimate of the magnitude of the distance traveled. If the classifier indicates that the agent has moved since the last measurement, we set the magnitude estimate to a constant value, typically the average cruising speed of the robot in practice. If it predicts no movement, a magnitude of zero is assigned. To ensure diversity and mitigate particle disparity, each particle’s motion update is individually calculated by incorporating zero-mean Gaussian noise into each measurement. The magnetometer is used to determine a heading, assuming an error within a fixed standard deviation. A simple odometry motion model from [23] is used. For magnitude calculations, we utilize the GMM likelihood to assign appropriate errors. Our testing revealed that when the classifier’s selection had a normalized likelihood of at least 0.9, the accuracy increases to 95%. When a selection has a high likelihood, we assign a lower sigma error to the magnitude; conversely, for less likely GMM classifications, a larger error term is applied to maintain particle diversity and ensure robustness.

After the motion update, new weights are computed for each particle. The updated weights are determined based on polled RSSI values from individual beacons. Broadcasting at regular intervals, each beacon’s RSSI values are recorded by the vehicle. The measured RSSI values are then compared with the modeled heat maps at the particles’ current position. To derive the weight values, we employ the inverse Mahalanobis distance between the collected RSSI values and the modeled RSSI values. By using the inverse distance, proximity between the measured and expected value corresponds to a smaller Mahalanobis distance, leading to higher

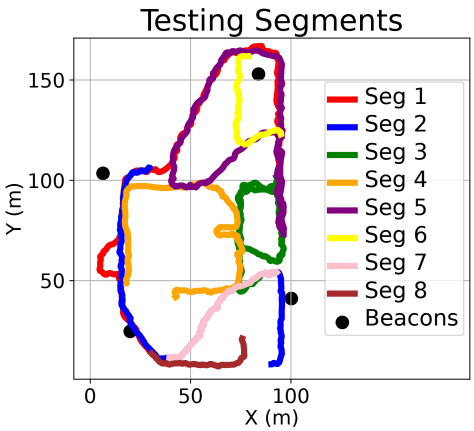

Fig. 6. Testing segments used for evaluation. Individual segments are excluded during the training of the GP or motion classifier. Segments 6-8 represent shorter paths where all adjacent prior data was removed to assess model generalization.

weight values for particles—indicating a stronger likelihood of accurate representation. At any given timestamp, we may receive RSSI values from independent beacons. This independence allows us to compute new overall particle weight values by multiplying the calculated weight values for each beacon. Upon updating every particle with its new weight, a normalization step is performed on the weights.

Finally, after the particles have been moved and new weights are calculated and normalized, the particles need to be resampled. Resampling is done with replacement according the new weights where particles with higher weights are more likely to be sampled. The entire particle filter process is then repeated at the next time step.

With each iteration, particles clustered around more probable regions are sampled with higher frequency, whereas those situated in less probable areas undergo less frequent sampling. This leads to the convergence of particles towards the regions deemed more likely. The belief state for the filter is determined by computing the weighted average of all particles, where the weights correspond to the assigned particle weights prior to resampling.

VI. Results and Discussion

We placed four LoRa beacons, mounted 2.5 meters above the ground across a portion of BYU campus with a receiving LoRa device mounted 1.25 meters atop an autonomous vehicle. Each beacon and receiver is powered by a Pycom LoPy4 board, operating in the ISM frequency of 915 MHz. Our vehicle is a custom low cost differential drive research platform as shown in Fig 1. To collect data, the vehicle drove around the test site for three hours, maintaining an average speed of 0.6 m/s. Ground truth was established using an RTK GPS, pairing positions with recorded RSSI values from the beacons, sampled at 1 Hz. Along with the RTK GPS and LoRa receiver, the vehicle is equipped with a magnetometer, low cost wheel encoders with four degree precision, and a non-RTK GPS.

From the collected data, eight test segments are partitioned and run to evaluate filter and GMM accuracy. Test locations can be seen in Fig. 6.

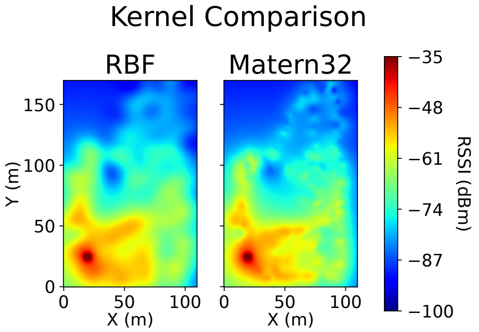

Fig. 7. Comparison of GP trained with RBF and Matérn32 kernels with similar training parameters. Notice the overfitting in the upper section of the Matérn kernel, leading to poorer generalization and accuracy.

A. Localization Technique Accuracy Evaluation

For each test segment, the testing portion is excluded from the training data for the GPs and the GMM. The first five test runs encompass longer routes where a majority of the path is retracing areas previously recorded. The subsequent three are shorter, consisting of unrecorded new areas. We will refer to these testing areas as “Prior Adjacent Data” and “No Prior Adjacent Data,” respectively.

We assess the efficacy of our GMM signal similarity classifier model by substituting alternative motion methods into the particle filter motion update step and comparing the resulting RMSE. The additional motion models include a constant velocity model and wheel encoder measurements. As shown in Table I where each experiment is repeated 50 times and errors are averaged, the motion update with wheel encoders yields the most accurate localization results with an RMSE of 2.12 meters. Notably, our model exhibits comparable accuracy solely using the LoRa RSSI measurements, averaging an RMSE of 2.57 meters. Both methods significantly outperform the naive approach of using the simplistic constant velocity model. When comparing our LoRa localization system with other conventional localization techniques, we observe that a standard GPS sensor averaged half a meter better in accuracy. All approaches significantly outperformed both dead reckoning wheel odometry and the weighted least squares method [1].

B. Kernel Evaluation

In contrast to previous works, which often leave the evaluation of kernels as an exercise to the reader, we provide the following experiment. During the training phase of our GP prior map, we substitute the RBF kernel with other common isotropic kernels and rerun all particle filter experiments. The results can be seen in Table II. We observe only a marginal improvement with the rational quadratic kernel over the RBF in regions lacking prior adjacent data. However, the RBF kernel exhibited a significant increase in accuracy in areas with prior adjacent data. As a result, we affirm that the RBF kernel adequately meets the requirements for our implementation.

Conceptually, the only difference between the kernels is an alpha parameter that can vary the lengthscale in different areas. This alpha parameter may overfit to noise or minor fluctuations when prior adjacent data exists, leading to worse performance compared to simpler kernels. On the other hand, in regions with sparse or no data, the flexibility of the rational quadratic kernel allows it to make more accurate predictions by capturing more complex relationships between input and output variables.

The flexibility of the Matérn kernel, with its characteristic peaks and steep drop-offs, can contribute to overfitting, especially in regions with sparse data. While high smoothness parameters enable it to capture intricate patterns, they may also cause excessive fitting of noise and outliers. Evidence of this is apparent in Fig. 7 where the Matérn kernel tends to overfit in areas that greatly differ from the mean function. We include combined kernels in our experiment but find that they do not substantially improve accuracy.

C. Generalization in Areas with No Prior Adjacent Data

To assess the generalization capabilities of our RSSI heat maps and particle filter, we evaluate the accuracy on three segments devoid of prior adjacent data. In these segments, any training points measured on the same path were removed. Segment 8, situated along the boundary of the model, lacks surrounding training data. Consequently, the GP heavily depends on its mean function for predictions, which hinders its ability to find the correct solution in this scenario. In contrast, segments 6 and 7 are situated along paths within the convex hull of training points, enabling the GP to interpolate effectively between these reference points with low error. When the GP can leverage prior surrounding data, the model finds an accurate solution as can be seen in Table III

D. Localization in GPS Denied Environments

To further investigate the generalization of our model and its robustness, we conduct an experiment within a partially obstructed GPS environment, devoid of any prior training data for the GP. The experiment entails the robot traversing a lap around a building, encompassing a covered walking space on both the west and east sides. The building can be seen in Fig. 8 b) and 8 c), with an approximated trajectory overlaid on the image.

Due to the unreliability of GPS signals along most of the trajectory, no ground truth is established, and the path is approximated. To obtain a rough estimate of error, we calculate the minimum L^2 distance to the path for each measurement. Running the particle filter experiment 50 times results in an RMSE error of 4.6 meters, while GPS yields an average RMSE of 26 meters, excluding the 10% of dropped measurements. The filter's estimate drifts to about 10 meters on the east side of the building, likely due to the lack of training data to model the signal fading next to the building. By integrating prior data gathered from beneath the overhang, this error could be significantly reduced. Although a rough estimate is used to approximate ground truth, this experiment demonstrates the resilience of our approach in scenarios where conventional methods such as GPS fall short.

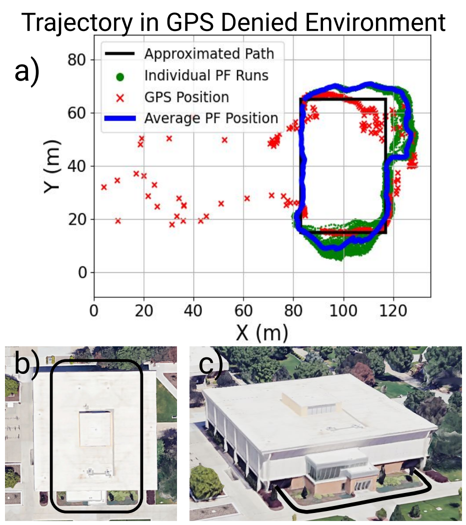

Fig. 8. Trajectory plots in a partial GPS denied environment. The particle filter is evaluated 50 times, with the estimate for each run shown in green. The blue line shows the average across all the runs while the red line represents the location tracked by the RTK GPS. The black line in each of the figures is the approximated path, as ground truth could not be established for most of the run due to the building overhangs as seen in c). Even in GPS denied environments our LoRa approach can still function. No training data was collected under the overhangs.

VII. Conclusion

This paper presents a method of localization based solely on RSSI measurements from LoRa modulating beacons and a magnetometer. The method involves modeling the RSSI of LoRa using GP regression, estimating motion via a novel probabilistic signal similarity classifier, and determining absolute localization through particle filtering. As a result, the proposed method achieves an accuracy of under three meters RMSE. Additionally, we believe this work marks the first known application of COTS LoRa technology for mobile agent localization.

We find that employing an RBF kernel for our GP yields the most favorable outcomes, demonstrating effective interpolation in test locations lacking prior adjacent data. When assessing various motion models for the particle filter, our motion classifier attains comparable accuracy to wheel encoders in estimating relative motion, but it can be applied to non-wheeled systems such as handheld devices or legged robots. Traditional GPS outperforms our magnetometer-LoRa localization system with an average accuracy of 0.5 meters RMSE, however in GPS-denied environments our method accurately estimates the robot location when GPS fails.

We plan to further investigate our method in GPS denied environments by substituting LoRa for underwater acoustic modems to localize marine agents. We are optimistic that our model can effectively interpret the inherently noisy signals and low broadcast rate characteristic of acoustic signals in shallow water.

TABLE I
EVALUATION RESULTS OF DIFFERENT MOTION MODEL AND LOCALIZATION TECHNIQUES WITH ERROR IN METERS

Evaluation Segment	PF Motion Type						GPS		Wheel Odometry	Weighted Least Squares
	Encoder		GMM (Ours)		Constant Velocity					
	RMSE	Std. Dev.	RMSE	Std. Dev.	RMSE	Std. Dev.	RMSE	Std. Dev.	RMSE	RMSE
1	**1.53**	0.96	**1.85**	1.10	2.83	2.23	2.07	0.88	53.20	134.11
2	**1.64**	0.92	**2.59**	1.26	5.21	2.80	2.25	1.23	71.63	87.47
3	**1.70**	0.66	**1.73**	0.81	4.39	2.69	1.98	0.78	100.32	48.16
4	**3.00**	3.45	**3.55**	3.26	5.33	3.25	1.90	0.77	30.27	109.56
5	**2.71**	1.25	**3.11**	1.84	5.29	3.84	2.13	0.91	54.72	77.44
Averages	**2.12**	1.45	**2.57**	1.65	4.61	2.96	2.07	0.91	62.03	91.34

TABLE II
GP KERNEL COMPARISONS ACROSS DIFFERENT TESTING ENVIRONMENTS WITH ERROR IN METERS

Kernel	No Prior Adjacent Data		With Prior Adjacent Data	
	RMSE	Std. Dev.	RMSE	Std. Dev.
Radial Basis Function	6.74	2.78	2.57	1.65
Rational Quadratic(RQ)	6.28	1.94	4.06	2.49
Matern 32	9.22	1.86	3.96	2.20
Sum(Matern, RQ)	9.17	3.88	3.99	2.49
Product(RBF, RQ)	12.29	3.12	5.90	3.38

TABLE III
RBF KERNEL EVALUATION WITH NO PRIOR ADJACENT DATA, WITH ERROR IN METERS

Evaluation Segment	RMSE	Std. Dev.
6	5.49	3.59
7	2.65	1.62
8	12.09	3.13
Average	6.74	2.78

REFERENCES

[1] P. Tarrio, A. M. Bernardos, J. A. Besada, and J. R. Casar, "A new positioning technique for RSS-Based localization based on a weighted least squares estimator," in *IEEE International Symposium on Wireless Communication Systems*, 2008, pp. 633–637.

[2] P. Tarrío, A. M. Bernardos, and J. R. Casar, "Weighted Least Squares Techniques for Improved Received Signal Strength Based Localization," *Sensors*, vol. 11, no. 9, pp. 8569–8592, 2011.

[3] S. Shang and L. Wang, "Overview of WiFi fingerprinting-based indoor positioning," *IET Communications*, vol. 16, no. 7, pp. 725–733, 2022.

[4] S. Boonsriwai and A. Apavatjrut, "Indoor WIFI Localization on Mobile Devices]," in *International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, 2013, pp. 1–5.

[5] B. Ferris, D. Hähnel, and D. Fox, "Gaussian Processes for Signal Strength-Based Location Estimation," in *Robotics: Science and Systems*, 2006.

[6] K. Ismail, R. Liu, A. Athukorala, B. K. K. Ng, C. Yuen, and U.-X. Tan, "WiFi Similarity-Based Odometry," *IEEE Transactions on Automation Science and Engineering*, pp. 1–11, 2023.

[7] S.-S. Choi, S.-H. Cha, and C. C. Tappert, "A Survey of Binary Similarity and Distance Measures," *Journal of Systemics, Cybernetics and Informatics*, vol. 8, pp. 43–48, 1 2010.

[8] S. Han, C. Zhao, W. Meng, and C. Li, "Cosine similarity based fingerprinting algorithm in WLAN indoor positioning against device diversity," in *IEEE International Conference on Communications (ICC)*, 2015, pp. 2710–2714.

[9] A. K. Shilpa Devalal, "LoRa Technology-an Overview," *International conference on Electronics, Communication and Aerospace Technology*, 2018.

[10] L. E. Marquez and M. Calle, "Understanding LoRa-Based Localization: Foundations and Challenges," *IEEE Internet of Things Journal*, vol. 10, no. 13, pp. 11 185–11 198, 2023.

[11] Junaedy, Alfin and Masuta, Hiroyuki and Sawai, Kei and Motoyoshi, Tatsuo and Takagi, Noboru, "Real-Time Simultaneous Localization and Mapping for Low-Power Wide-Area Communication," in *IEEE Symposium Series on Computational Intelligence (SSCI)*, Dec 2020, pp. 1905–1912.

[12] Alfin Junaedy and Hiroyuki Masuta and Kei Sawai and Tatsuo Motoyoshi and Noboru Takagi, "LPWAN-Based Real-Time 2D SLAM and Object Localization for Teleoperation Robot Control," *Journal of Robotics and Mechatronics*, vol. 33, no. 6, pp. 1326–1337, 2021.

[13] Hu, Kang and Gu, Chaojie and Chen, Jiming, "LTrack: A LoRa-Based Indoor Tracking System for Mobile Robots," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 4, pp. 4264–4276, April 2022.

[14] Z. He, Y. Li, L. Pei, and K. O'Keefe, "Enhanced Gaussian Process-Based Localization Using a Low Power Wide Area Network," *IEEE Communications Letters*, vol. 23, no. 1, pp. 164–167, 2018.

[15] Delafontaine, Victor and Schiano, Fabrizio and Cocco, Giuseppe and Rusu, Alexandru and Floreano, Dario, "Drone-aided Localization in LoRa IoT Networks," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2020, pp. 286–292.

[16] E. Schulz, M. Speekenbrink, and A. Krause, "A Tutorial on Gaussian Process Regression: Modelling, Exploring, and Exploiting Functions," *Journal of Mathematical Psychology*, vol. 85, pp. 1–16, 2018.

[17] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2005.

[18] M. Kanagawa, P. Hennig, D. Sejdinovic, and B. K. Sriperumbudur, "Gaussian Processes and Kernel Methods: A Review on Connections and Equivalences," 2018.

[19] E. Bingham, J. P. Chen, M. Jankowiak, F. Obermeyer, N. Pradhan, T. Karaletsos, R. Singh, P. A. Szerlip, P. Horsfall, and N. D. Goodman, "Pyro: Deep Universal Probabilistic Programming," *J. Mach. Learn. Res.*, vol. 20, pp. 28:1–28:6, 2019.

[20] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Cambridge University Press, 2024.

[21] A. Ranganathan, M.-H. Yang, and J. Ho, "Online Sparse Gaussian Process Regression and its Applications," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 391–404, 2011.

[22] K. Murphy and S. Russell, "Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks," in *Sequential Monte Carlo methods in practice*. Springer, 2001, pp. 499–515.

[23] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA: The MIT Press, 2005, ch. 5.4 Odometry Motion Model, pp. 107–111.